Developing a cooperative robotic-assisted surgical system for endoscopic endonasal surgery

Jacinto Colan

To my loving parents.

Acknowledgements

This dissertation would not have been possible without the help of so many people in so many ways. I would like to express my sincere thanks to all of them.

First and foremost, I would like to express my deep gratitude and thanks to my advisor, Prof. Hasegawa, for accepting me as his student, his endless support, and mentorship. The door to Prof. Hasegawa's office was always open whenever I ran into a trouble spot or had a question about my research. I am also grateful to Prof. Nakanishi for his unwavering support, guidance, and insight throughout the development of this research. He consistently helped me improve my work with his thoughtful comments that pushed me to sharpen my thinking and brought my work to a higher level.

I am highly indebted to the members of Hasegawa's lab, who have helped me in countless opportunities since my arrival to Japan. To Prof. Sekiyama, Prof. Aoyama, Prof. Takeuchi for their continuous assistance at every stage of my research. To the former members of my research team, Sato-san, Okumura-san, Stanley, Ohara-san, and Uozumi-san, with whom I shared so many experiences working together, kindly assisted me with Japanese and were always willing to help me. To my labmates, Noel, Rubens, Liu, Li, Zhou, and many others, for all these years full of talks, fun, and friendship.

I would like to thank the Japanese Ministry of Education, Culture, Sports, Science and Technology for their financial support through the MEXT Scholarship during the entire duration of my post-graduate studies.

I am extremely grateful to my family and friends for their encouragement, understanding, and support, which helped me in the completion of this work. To my partner Ana, thanks for your patience, support, and caring over all these years.

Finally, I must express my very profound gratitude to my dear parents, for their significant role in my life, unfailing support, and continuous encouragement throughout all my years of study.

This accomplishment would not have been possible without them. Thank you.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

1.1.1 Endoscopic endonasal surgery

The pituitary is a pea-sized gland located at the base of the brain, behind the eyes and attached to the hypothalamus via nerve fibers, as shown in Figure 1.1. It produces critical hormones that help to regulate other endocrine glands such as the adrenal and thyroid. Abnormal growth of pituitary adenomas and Rathke's cleft cyst can be developed in the pituitary gland or in closed proximity. Pituitary tumors are usually noncancerous and slow-growing. However, they can alter the functioning of the pituitary, causing an excessive or insufficient production of hormones. In the case of large tumors, compression of the surrounding areas to the optic nerves can also produce a total or partial loss of the patient's vision [1]. In these cases, surgery is required to remove the tumor.

A transsphenoidal approach was proposed by Cushing in the early 1900s, to remove pituitary tumors of more than 200 patients [3]. Cushing's original method consisted of a sublabial septal incision to reach the nasal septum and expose the sphenoid sinus. Compared with traditional open approaches, either transcranial or transfacial that require traumatic access through the patient's forehead or cheek, the transsphenoidal approach significantly reduced invasiveness and mortality rate. Despite the encouraging results, the technique was abandoned until the mid-1950s, where the use of microscopic vision and instraoperative fluoroscopy permitted a more effective and safer resection of pituitary tumors. These modified methods are still widely used by neurosurgeons today. However, the lack of a panoramic view inside the sphenoid cavities and the sella turcica remains an important limitation.

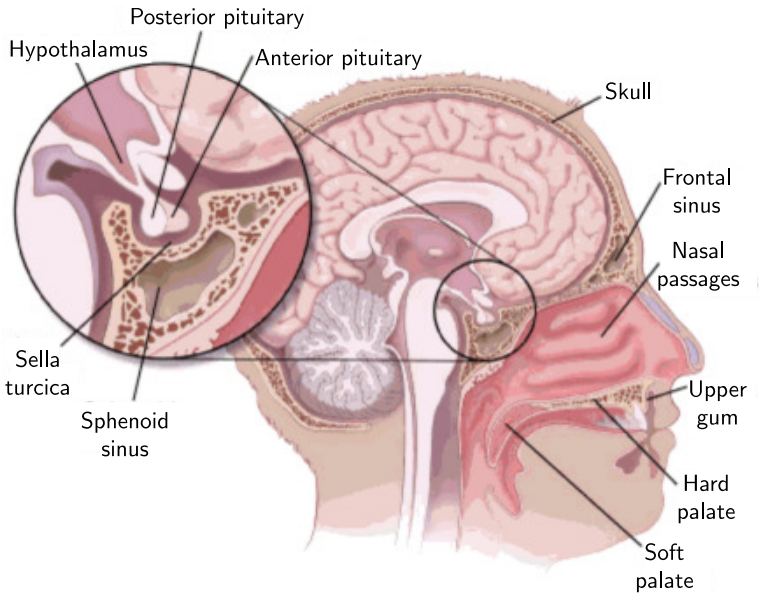

Figure 1.1: The pituitary gland is located at the base of the brain and attached to the hypothalamus. (Reprinted with permission from the American Cancer Society [2])

In 1992, Jankowski et al. [4] described the first successful removal of pituitary adenomas in three patients using a purely endoscopic endonasal approach. Endoscopic endonasal surgery (EES) is a mononostril or binostril technique performed via the insertion of an endoscope and long surgical instruments into the sphenoidal sinus cavity by going through the nose. Typically, this procedure uses a 0deg or 30deg endoscope and surgical instruments with about 20cm length and 4 mm wide. Conventional surgical instruments used for EES are hand-held instruments with long shafts, an end-effector (needle holder, dissector, etc.) at one end, and a handle at the other. EES has benefited from the development of new technologies, including advanced endoscopic cameras and navigation systems. The modern neuroendoscope provides excellent illumination with a wide panoramic view, 3D images, and HD video quality. Since Jankowski's results, EES has become a well-accepted and widely used procedure for the removal of pituitary adenomas and tumors at the skull base [5]. Several authors [6],[7] have compared patients who underwent tumor resection of pituitary adenomas with either endonasal endoscopic or a microscopic transsphenoidal approach and demonstrated a significantly higher rate of total resection, less mean blood loss and fewer nasal complications within the endoscopic group.

1.1.2 Endoscopic endonasal surgery limitations

Endoscopic surgery can reduce the patient's trauma, but is limited in maneuverability and haptic feedback compared to open surgery where the surgeon can directly view and touch the surgical workspace. The surgeon's ability to visualize and dexterously manipulate the surgical target is limited by working with long, rigid instruments (see Figure 1.2a) inserted through the nasal cavity over which the instrument is constrained with a reduced range of motion to four degrees of freedom (DOFs): translation along the shaft of the instrument; rotation around the translational axis; and limited inclination of the shaft pivoted trough the nostril. As a result, some areas at the skull base would be difficult or almost impossible to reach.

Hand-eye coordination is another major challenge with a two-dimensional endoscopic view that is not aligned with the surgeon's axis of view and mirrored movements of the shaft of the surgical instruments against the surgeon's hand motion. Besides, the mobility of the endoscope creates potential conflicts with other instruments inserted through the nasal cavity (see Figure 1.2c), taking up valuable space. As a consequence, surgeons experience a significant steep learning curve to master EES techniques [8].

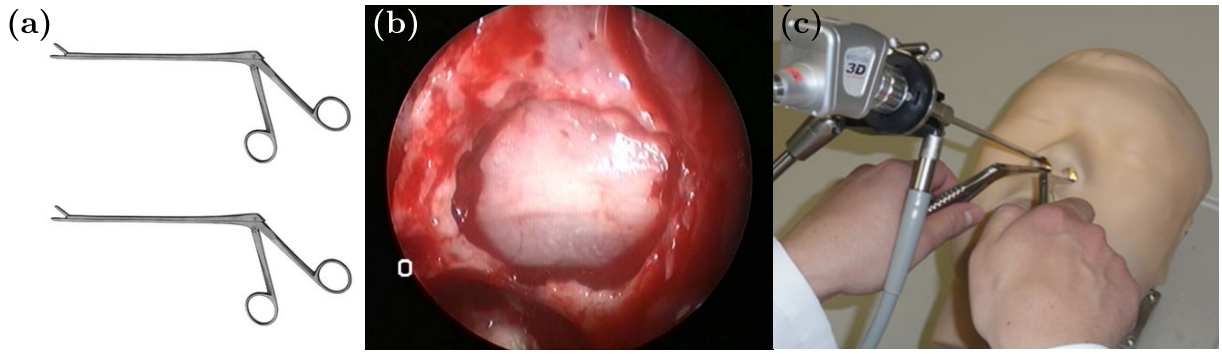

Figure 1.2: Challenges in EES. **(a)** Conventional surgical instruments for EES. **(b)** 2D endoscope view (Reprinted with permission Kenan et al. [8]). **(c)** Constrained workspace for multiple instruments (Reprinted with permission from Kawanishi et al. [9]).

Because of the reduced workspace and the lack of direct visualization, the execution of complex surgical tasks increases surgeon's work effort. Suturing is considered one of the most challenging and time-consuming tasks in minimally invasive surgery [10], and is a common procedure in EES to secure the reconstructed dura after endonasal tumor resection. Inadequate suture can lead to cerebrospinal fluid leakage, which is a common postoperative complication in EES. Furthermore, the two-dimensional endoscopic (see Figure 1.2b) view makes it difficult to recognize the relative position of the tissue from the

surgical instruments, resulting in multiple attempts and increased tissue trauma to achieve a proper suture [11].

1.2 Robotic-assisted surgical systems

Current advances in the development of robotic-assisted surgical systems have proven to be effective in reducing patient trauma and hospitalization time by preventing the risk of complications [12]. Robotic systems provide several advantages that can be exploited in surgery applications such as repeatability, high accuracy and remote operation. Challenges encountered in traditional surgical procedures have also inspired developments in technologies for visual perception, enhanced dexterity, transparent eye-hand coordination, and haptic sensing, providing surgeons with superhuman capabilities. They can also help to reduce the steep learning curve for surgeons associated with minimally invasive surgery, providing automatic control of the endoscope camera, using intelligent instruments or assisting in surgical procedures with higher dexterity [13].

The cooperative role of robotic systems in surgical procedures can be divided into five categories [14]:

- Assistant replacements: The robotic system performs auxiliary assistive tasks, usually performed by surgical interns, e.g., camera positioning.

- Telesurgical systems: The robot's motion is commanded by the surgeon by means of a control interface, e.g. a joystick. The main benefits of such systems is the ability to perform surgical procedures remotely. They also can reduce the hand tremor for microsurgical tasks and access to areas difficult to reach through high-dexterity tools.

- Navigational aids: These systems provide accurate positional information about the location of surgical instruments relative to the patient's anatomy. They usually comprise a 3D localization device, such as a 3D optical tracker, and a graphical interface for displaying the device position relative to volumetric medical images.

- Precise positioning system: These systems act as a fixed frame to guide a tool (e.g., biopsy needle) to a desired position and orientation relative to the target anatomy. They can provide higher flexibility compared to passive systems, by rapidly readjusting the target pose for a different application. Once the pose is fixed, they are turned off to insert the tool.

- Precise path systems: The robot is moved through a defined path to complete a specific task. For example, a precise machining of bone either using the robot to move the cutting tool or as a means of constraining the surgeon to keep the tool within a predefined volume.

Each surgical domain, e.g., neurosurgery or laparoscopic surgery, imposes its own requirements and constraints. Consequently, many robotic systems have been proposed for specific surgical scenarios with different human-robot interface designs and assistance approaches. The classification scheme used on this work is based on the cooperative characteristics provided by the robotic surgical system, distinguishing between three main categories: active, semi-active, and synergistic [15]. Synergistic systems can also be divided into three subgroups: handheld, hands-on, and telemanipulated.

1.2.1 Active systems

Active systems provide full autonomy during the execution of a surgical task. The robot carries out a surgical procedure by executing a set of instructions predefined by the surgeon. The surgeon supervises the execution of the task and can manually interrupt the procedure in case of an emergency (see Figure 1.3).

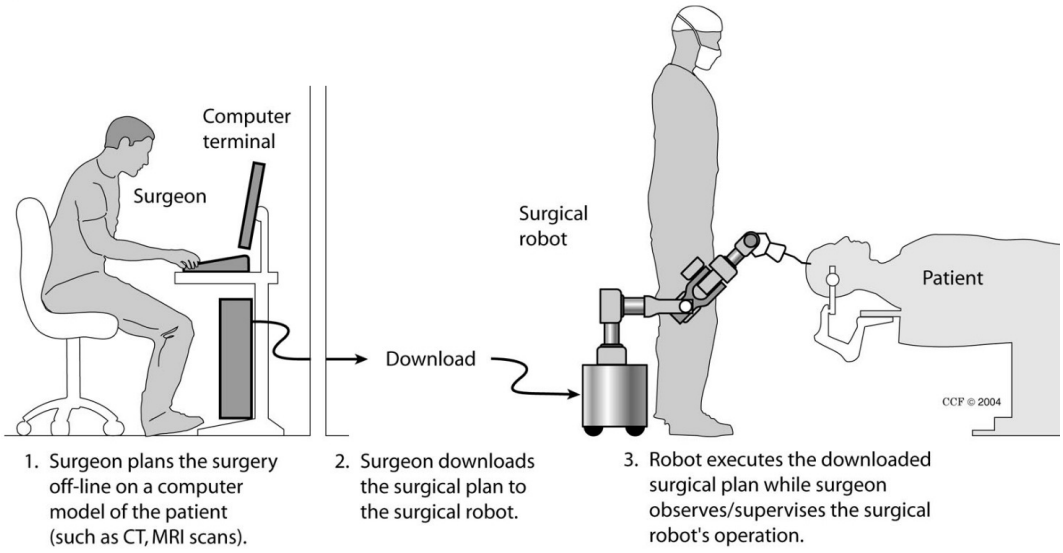

Figure 1.3: Conceptual schematic of active robotic systems (Reprinted with permission from Nathoo et al. [16]).

ROBODOC (Integrated Surgical Systems, Davis, CA), shown in Figure 1.4, was the first active surgical system developed by Taylor et al. [17] for orthopedic surgery, and

later further developed for hip replacement procedures [18]. It consists of three main components: a planning workstation, the robot itself, and a workstation that controls the robot motion. However, the surgical community has been reluctant to adopt autonomous systems in surgery as the surgeon wishes to be in control of the operation rather than only an observer.

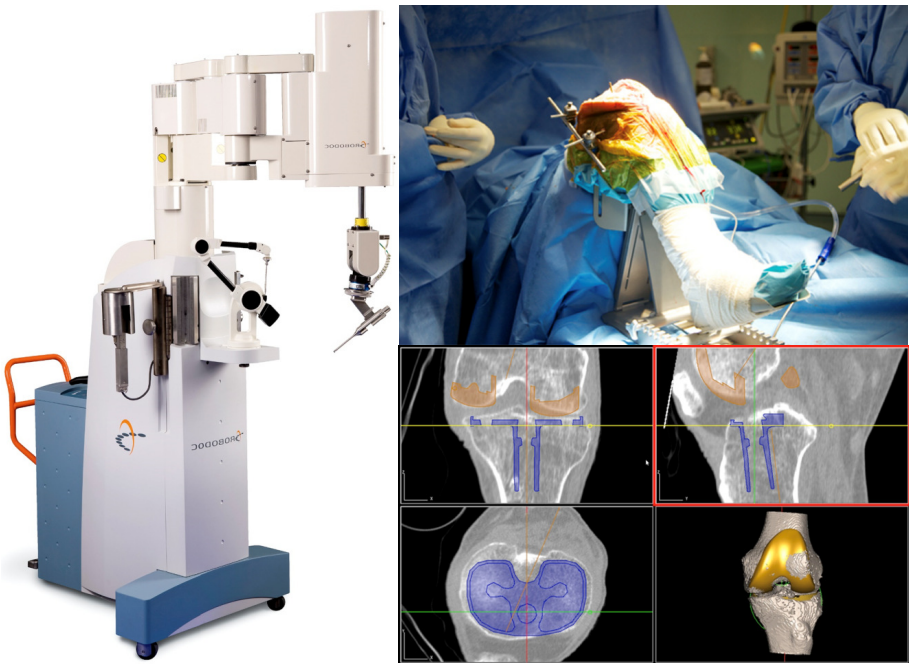

Figure 1.4: The first autonomous surgical system, the ROBODOC used for orthopedic surgery (Reprinted with permission from Liow et al. [19]).

1.2.2 Semi-active systems

Semi-active systems work as mechanical tool guides, physically constraining the surgeon command into a predefined trajectory. The robot position is fixed to steer a tool operated by the surgeon, reducing the number of DOFs to be controlled. Example of applications are tool positioning or alignment, e.g., biopsy extraction, laser-guided alignment, and stereotactic neurosurgery [20].

1.2.3 Synergistic systems

Synergistic systems provide shared control between the surgeon and the robot in which the position and the movement of the surgical instrument are controlled by a master

device commanded by the surgeon. Unlike active and semi-active systems, synergistic systems do not require a pre-operative plan to be followed. Such synergistic systems have been extensively developed during the last few decades and can be sub-classified into teleoperated, hand-held, and hands-on devices.

Teleoperated systems

Teleoperated robotic systems emerged in the seek of increasing dexterity of the minimally invasive surgery inside small cavities. These systems are designed to provide surgical dexterity through stereoscopic vision, dexterous end-effectors, and tremor filtering. In these systems, the surgeon performs the surgical manipulation through an online input interface that is typically a joystick, and the robotic manipulator follows faithfully the motions of the input interface (see Fig. 1.5).

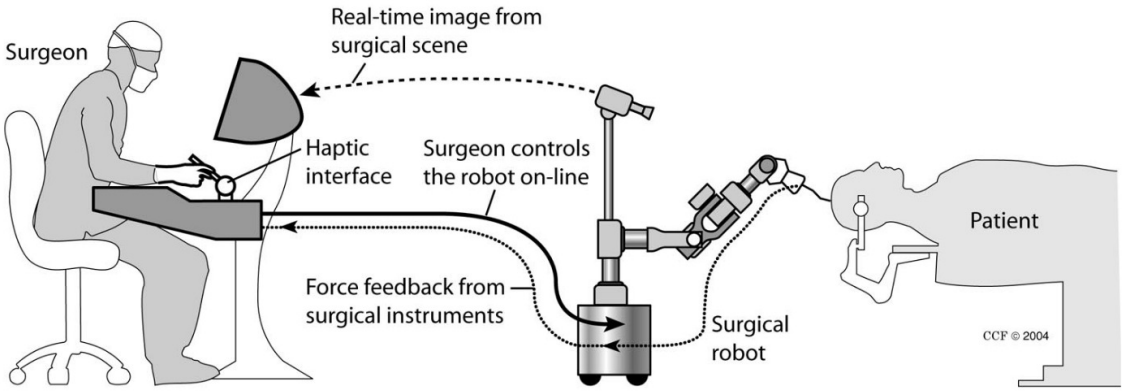

Figure 1.5: Conceptual schematic of teleoperated robotic systems (Reprinted with permission from Nathoo et al. [16]).

Teleoperated robotic systems have been developed mostly for laparoscopic applications. The commercially available da Vinci Surgical System [21], shown in Fig. 1.6, is a notable example of such an approach, which consists of the control console, a control unit, and a three-arm surgical manipulator. It provides three-axis wrist tools which resemble the degrees of freedom of a human wrist, a 3D visual display through an stereo endoscope, and scaling and filtering of involuntary motions caused by human hand tremor.

Figure 1.6: daVinci robotic system (Reprinted with permission from Di Maio et al. [22]).

In general, teleoperation interfaces can provide additional capabilities to surgeons such as ergonomics, enhanced vision (3D stereoscopic visualization), motion scaling, haptic feedback, and tremor suppression. Depending on their kinematic design, teleoperation interfaces are mainly based on serial mechanisms (e.g. PHANTOM [23]) or parallel mechanisms (e.g. Sigma7 [24]) as shown in Fig. 1.7.

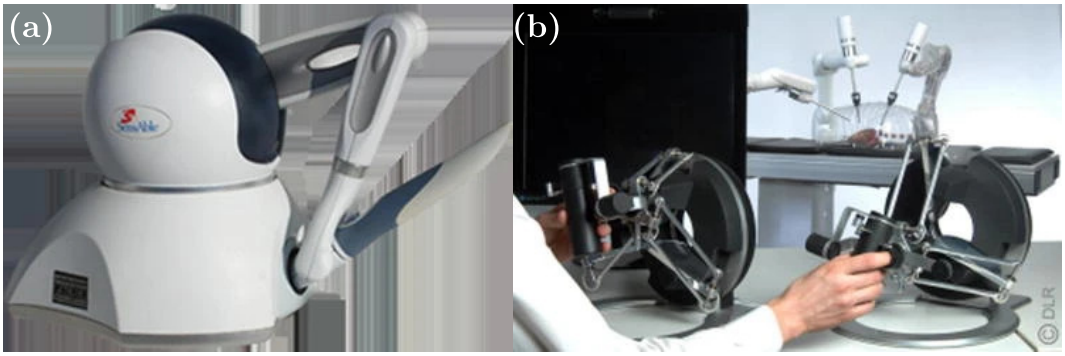

Figure 1.7: Teleoperation haptic devices. **(a)** Phantom Omni. **(b)** Sigma 7. (Reprinted with permission from Simorov et al. [25]).

Hand-held systems

Hand-held instruments provide direct control over surgical instruments and natural response and haptic feedback. They are significantly smaller and not fixed on a ground frame, designed to exploit the surgeon's dexterity with the use of ordinary surgical instruments. Enhanced features include tremor suppression and active guidance. Robotic

hand-held instruments have been used in laparoscopic surgery [26, 27], eye surgery [28], and microsurgery [29]. Optical trackers are typically used to obtain the instrument's pose in the space, which is one of the major drawbacks. Such optical trackers can be easily occluded during a surgical task and represent a challenge for applications in minimally invasive surgery. Furthermore, the lack of a ground frame can produce undesired reaction forces on the operator's hand [30].

The main issue in the design of a hand-held robotic surgical device is related to the mapping between the DOFs of the surgeon's hand and the end-effector's DOFs and how the surgeon controls the end-effector. One approach is using a finger-operated handle, where the end-effector is controlled by using buttons, dials, or joysticks integrated in the handle as in [31], [26]. A different approach is the use of an articulated handle, where the articulation's DOF of the handle can be mapped to the DOF of the end-effector. In minimally invasive surgery, the surgeon has to perform a cognitive remapping to resolve the incompatibility of the viewpoint presented by the endoscope and surgeon's spatio-motor expectations. A nonintuitive control mode makes this remapping more difficult, leading to long learning curves, longer operation times, and additional burden on the surgeon. Between hand-held devices, we can distinguish mechanical and robotic instruments.

Mechanical articulated hand-held instruments

Mechanical hand-held instruments have an articulated handle, rotating knobs or similar mechanical controllers, and a mechanical transmission system to actuate two or three DOFs of the end-effector. They are characterized by a functional mechanical design and low cost, but often lacks of an intuitive interface. The most well-known devices are RealHand [32] (see Figure 1.8) from Novare Surgical Systems, and Laparo-Angle from Cambridge Endoscopic Devices [33] (see Figure 1.8).

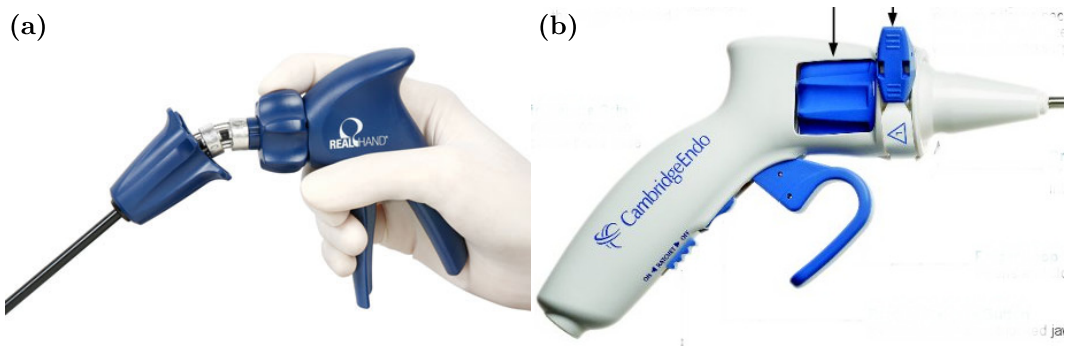

Figure 1.8: Mechanical hand-held instruments. **(a)** RealHand. **(b)** Laparo-Angle

Realhand has a wrist added to the end-effector, so that it can allow yaw and pitch movements, making the total number of DOF of the instrument six. Its handle is designed similar to the conventional pistol-grip handles and articulated as well to control the additional DOF. When the handle is bent, the end-effector bends in the same direction. The articulation between the handle and the shaft is a universal joint so that rotating the handle makes the shaft of the instrument rotate. Laparo-Angle has an articulated wrist and an articulated handle. But its end-effector has one more DOF compared to RealHand: the distal tip can turn 360 degrees at any angle using an axial rotation knob in the handle. The handle of Laparo-Angle has a more ergonomic design. The transmission mechanism is cable driven and the bending structure is made of a stack of interrelated links driven by four cables. The handle is designed like a lever under the middle fingers and its up/down movements correspond to the distal tip's up/down movements. Table 1.1 shows some of the hand-held instruments available on the market and list the differences between them in terms of kinematics and control. While they all claim to be intuitive and dexterous, most surgeons would still prefer using classic instruments.

Table 1.1: Mechanical Hand-held instrument comparison. Reprinted with permission from Zahraee et al. [27]

Feature	RealHand	Laparo-Angle	Roticulator	Radius
Kinematics	Y-P	Y-P-R	Y-R	Y-P
Controllers	articulated handle	articulated handle, knob	knobs	articulated handle
End-effector lock	No	Yes	Yes	No
Need use of other hand	No	For lock	For yaw	No
Shaft rotation	by a shaft screw	by rotating the handle	by rotating the handle	by a shaft screw

Robotic hand-held instruments

In robotic hand-held devices, the manual control and actuation system are replaced with an electronic controller and electrical actuators. One example of this mechanism is the instrument developed by Jinno et al. [31], which can perform roll-yaw movements and

the opening/closing of the grasper. The electric motors are located on the shaft of the instrument. The transmission mechanism is cable driven and a combination of gears makes a distal mechanism that can perform yaw-roll movements and at the same time open/close the grasper. The handle has a cylindrical shape and is positioned like a pistol-grip handle in 90 degrees to the shaft. A button and two knobs on the handle allow opening/closing of the grasper, deflecting the distal tip or turning it around its axis. The processor unit is away from the instrument and connected to it through an electrical cable. In the instrument developed by Piccigallo et al. [26], the motors and controller are all separated from the instrument (see Figure 1.9b). A Bowden cable actuation system with eight pretensioned cables transmits forces of the motors to the pulley box on the instrument's shaft. The end-effector has a Roll-Pitch-Roll kinematics. The handle is of cylindrical shape. However, it is not rigidly connected to the shaft. The pitch axis between the handle and the shaft allows the user to lower his arm and keep a comfortable pose. In [34] the authors developed a 5 mm instrument composed of four parts: an ergonomic handle, a shaft with a multi-DOF distal tip, an active trocar, and a controller. Figure 1.9a shows the the prototype developed in [34]. The instrument has a total of six DOFs of which three DOFs are manual and the other three, i.e., rotating the shaft, bending and rotating the distal tip, are robotic. To use the instrument, the active trocar is plunged on top of a normal trocar.

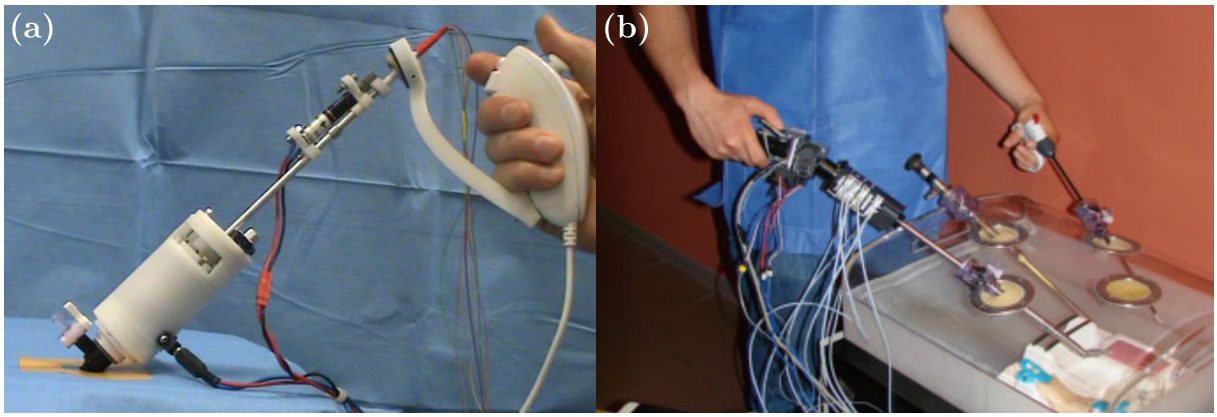

Figure 1.9: Robotic hand-held devices. **(a)** Robotic hand-held device (Reprinted with permission from Hassan-Zahraee et al. [34]). **(b)** Robotic hand-held device (Reprinted with permission from Piccigallo et al. [26]).

Hands-on systems

In hands-on systems, both a human and a robot manipulator are in control of a surgical instrument by direct contact as shown in Figure 1.10. In such a scenario, the surgeon's ex-

perience, intuition, and reaction ability are combined with the high accuracy and reliability provided by the robotic system. This approach can offer advantages in the context of micromanipulation compared with teleoperated systems such as: simplicity, direct contact and coupling with natural senses, less expensive implementation, straightforward integration with existing environments and great immediacy for the surgeon. The main disadvantages are the loss scaling and the ability to manipulate objects remotely.

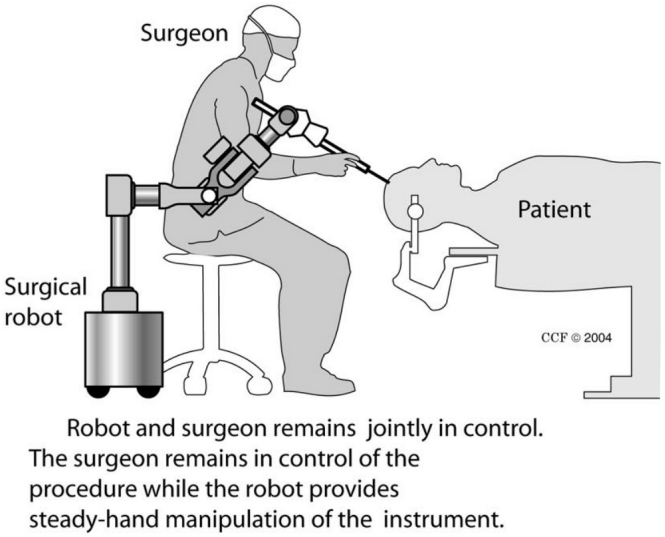

Figure 1.10: Conceptual schematic of hands-on systems (Reprinted with permission from Nathoo et al. [16]).

This approach is followed in the Acrobot system [35] (see Fig. 1.11a) and the MAKOplasty system [36] (see Fig. 1.11b) developed for orthopedic applications with active constraints limiting the instrument motion inside prescribed anatomic boundaries. The Steady-Hand robot developed by Taylor et al. [37] for eye surgery is another example (see Figure 1.11d), where force augmentation with steady manipulation is provided through active constraints to limit the motion within predefined regions. PADyC, shown in Figure 1.11c, is another example of hands-on surgical system proposed for cardiac puncturing [38].

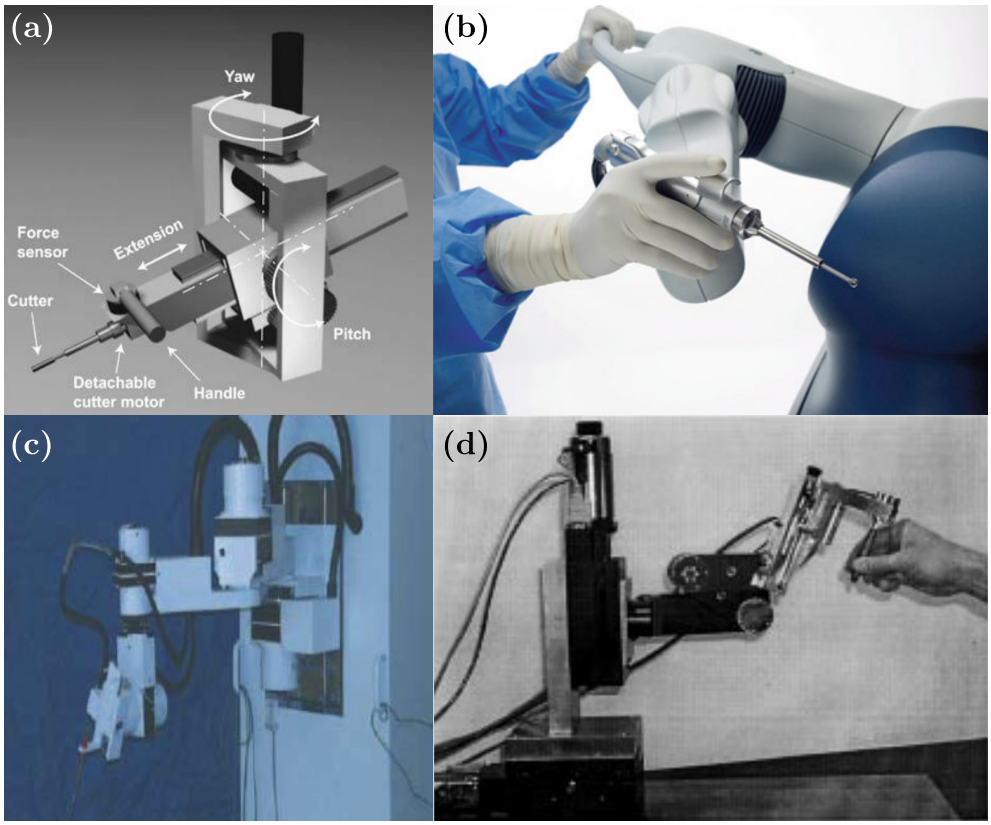

Figure 1.11: Hands-on surgical systems. **(a)** ACROBOT (Reprinted with permission from Jakopec et al. [35]) **(b)** MAKOplasty (Reprinted with permission from Hagag et al. [36]) **(c)**PadyC (Reprinted with permission from Zhan et al. [39]) **(d)** The Steady-Hand robot (Reprinted with permission from Taylor et al. [37])

1.3 Related works

1.3.1 Robotic-assisted surgical systems for EES

Robotic systems can enhance the surgeon's capabilities with improved precision and safety during the operation, improving ergonomics, and decreasing the surgeon's effort. However, EES remains a challenge in surgical robotic applications [40] with few systems targeting EES.

Teleoperated systems have been used with limited success. Attempts have been conducted with the da Vinci system in a transoral [41] and transantral [42] approach. While the use of highly dexterous instruments can provide operation benefits, both approaches generate excessive trauma and remain highly invasive due to the size of surgical instru-

ments. In [43], a continuum robot teleoperated by two PHANTOM devices was proposed for transnasal surgery (see Figure 1.12a). However, it required a large workspace and a complex setup procedure. Additional attempts shared similar limitations with the size of the system and instruments, time-consuming setup procedure, and the considerable cost of these systems. Importantly, teleoperated systems limit the surgeon's physical access to the patient. Thus, the help of bedside staff would be needed for the use of auxiliary instruments or the change of instruments, which is a frequent task in EES surgeries.

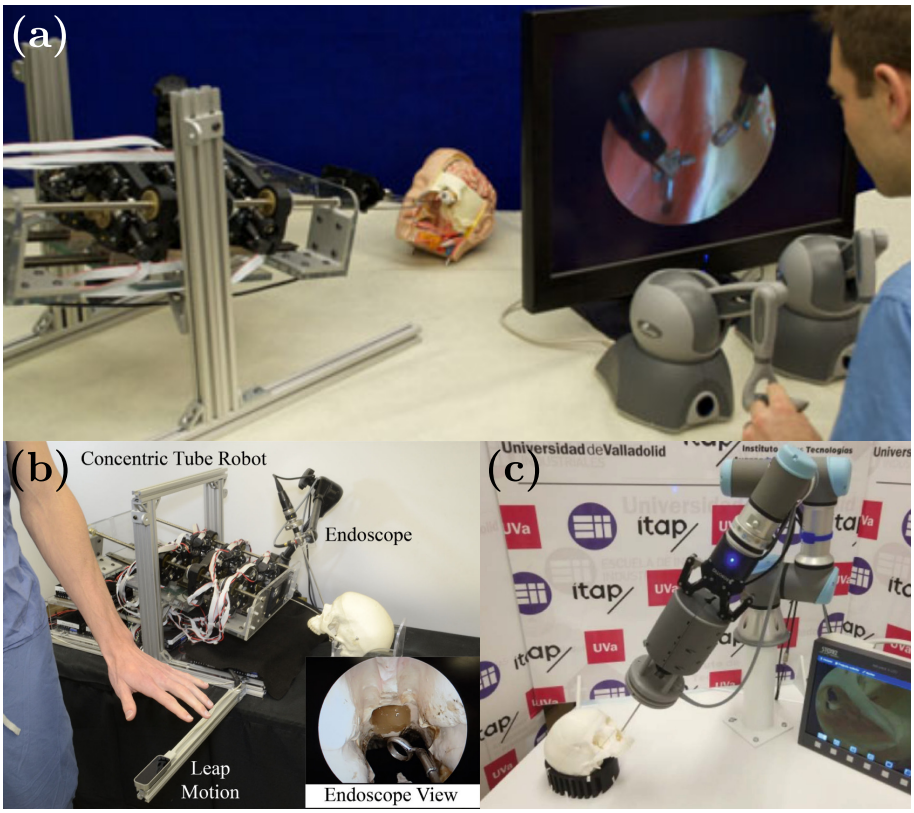

Figure 1.12: Surgical systems for EES. **(a)** Continuum robot (Reprinted with permission from Burgner et al. [43]) **(b)** Leap Motion (Reprinted with permission from Travaglini et al. [44]) **(c)** CRANEEAL project (Reprinted with permission from Garcia et al. [45])

Additional teleoperation interfaces for EES include a hand tracking device (Leap Motion) used to control a continuum robotic system [44] (see Figure 1.12b), and a voice-controlled robot for holding and maneuvering an endoscope (AESOP) [46]. However, differences in the coordinate frames between the surgeon and robot combined with limited communication channels make it difficult to control active tools precisely.

In the case of hands-on systems, no specific implementation for EES was found, but some systems have been proposed for neurosurgery tasks, such as commanding a drill with

a modified version of the Steady-Hand robot [47] and the NeuroMate robot [48], and to control the position of the endoscope using the interaction force and virtual fixtures in a pre-planned operation [49]. The CRANEEAL project conceptualized a similar collaborative approach for controlling a robot arm in EES [45] (see Figure 1.12c).

1.3.2 Cooperative manipulation and shared control

A critical issue in the development of cooperative robotic systems is the adaptability of the robot to task conditions and environmental constraints. Hogan [50] proposed an impedance control framework to model the dynamic relationship between the robot and environment. Although the original work does not cover the case of human-robot interaction, it has been further explored in following works by Kozuge and Kazamura [51], Tsumugiwa et al. [52], and Gribovskaya et al. [53]. In [51], several control strategies were proposed to allow the human and robot manipulator to carry heavy and bulky objects. The robot is responsible for carrying a payload and the human steers the object to a desired position or along a desired path. In [52], the authors use virtual forces to augment the sensor space and perform a collaborative peg-in-hole task.

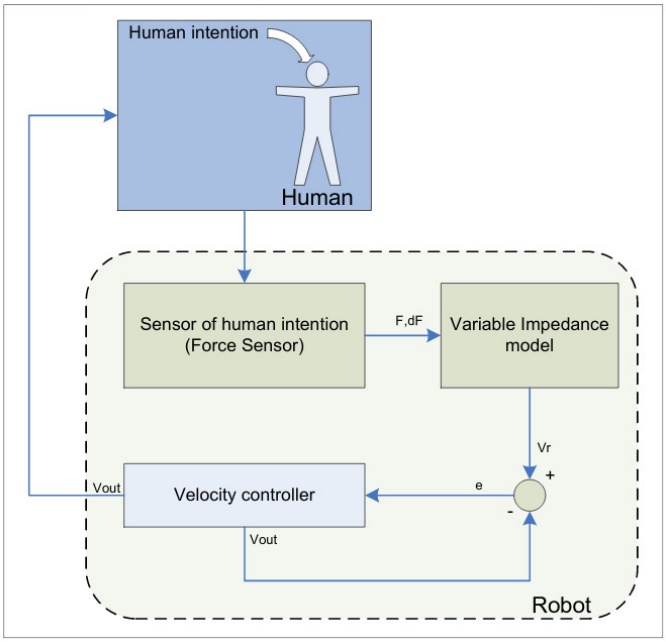

Figure 1.13: Human-robot cooperation scheme proposed by Duchaine et al. (Reprinted with permission from Ducahine et al. [54])

Force/pressure sensors are usually used in the design of human-robot interfaces to recog-

nize the contact and anticipate the desired action from the human partner or to control the cooperation effort. This approach has been explored in several applications such as collaborative object transportation [51],[52],[55],[54], object lifting [56], object placing [52],[57], object swinging [58], posture assistance [59],[60], and industrial complex assembly processes [61]. In most of these works, the interaction forces/torques are used to control the robot parameters and trajectories following the admittance [54] or impedance [52] causality.

However, in spite of the multiple applications, collaborative tasks involve interaction with an uncertain environment that can generate unpredictable force components in the sensor data. In complex and dynamic scenarios, this can significantly reduce the suitability of the interface to distinguish between the commands given by the operator and the forces generated from the interaction with the environment.

In cooperative surgical systems, the surgeon and robot share control to perform a surgical procedure. One of the first surgical applications with robotic assistance was in stereotactic neurosurgery [62], in which the robot autonomously positions and fixes a mechanical guide according to the preplanned trajectory, and then the surgeon uses this guide to introduce the surgical tool such as a drill, a probe, and an electrode. Kwoh et al. [63] made the first attempt to use an industrial PUMA 560 robot for CT-guided brain tumor biopsies. In the late 1980s, Benabid et al. developed an early precursor to the stereotactic robot called NeuroMate [64] (see Figure 1.14). This system has been used clinically in over 1600 procedures since 1989.

Figure 1.14: Current version of Neuromate robotic system used for stereotactic surgery. (Reprinted with permission from Zanotto et al. [65])

Early experiences with surgical robots, such as ROBODOC and other similar systems, showed that surgeons found a form of hands-on control to be very convenient and natural

for surgical tasks. The robot provides a steady-hand manipulation of surgical instruments while the surgeon control the whole procedure.

1.3.3 Spatial Motion Constraints

A virtual fixture, also known as virtual mechanism or virtual tool, is a software-generated constraint on the manipulator motion imposed on a human operator via robotic devices. It helps to perform manipulation tasks in collaboration with a robotic system by limiting movement into restricted regions or facilitating the movement along desired paths. A non-desired direction of movement can be constrained by avoiding any command over that direction.

Depending on the application and workspace, multiple virtual fixtures can be implemented in the control strategy. Ho et al. [66] distinguished between two approaches which they called the implicit force control and the modified damping control. In the former, no force sensor is used. The latter approach uses a force sensor to measure operator's guiding force, in order to determine the robot's velocity. In this case, the robot's desired velocity is controlled based on the relative position of the robot, the motion constraint, and the direction and magnitude of the guiding force. It makes use of admittance control techniques and can be applied to robots of the admittance type in a very natural way.

Based on the Steady-Hand Robot, Bettini et al. [67], uses admittance control to develop guidance virtual fixtures to assist the surgeon to move the surgical instruments in the desired direction (see Figure 1.15). Their work focused on 2D geometric guidance motion of the tool tip based on vision information. Funda and Taylor [68] formulated desired motions as sets of task goals in any number of coordinate frames relevant to the task, and optionally subject to additional linear constraints in each of the frames for redundant and deficient robots. Li et al. [69] extended Funda's work to generate virtual fixtures for real-time obstacle avoidance, and simultaneously to assist the surgeon to perform the desired tool motion to accomplish the intended tasks by using an optimization-based approach.

Virtual fixtures have also been widely applied to telemanipulators, where a human operator manipulates a master robotic device, and a remote slave robot manipulates an environment while following the commands of the master. Rosenberg [70] implemented virtual fixtures as an impedance surface on the master to assist in peg-in-hole tasks. Joly et al. [71] simulate a virtual mechanism connected to the master and slave arms via springs and dampers to impose motion constraints on the system.

Implementation of virtual constraints has been explored to help a surgeon to perform several tasks such as: alignment of a tool, following a precise path, maintaining a desired

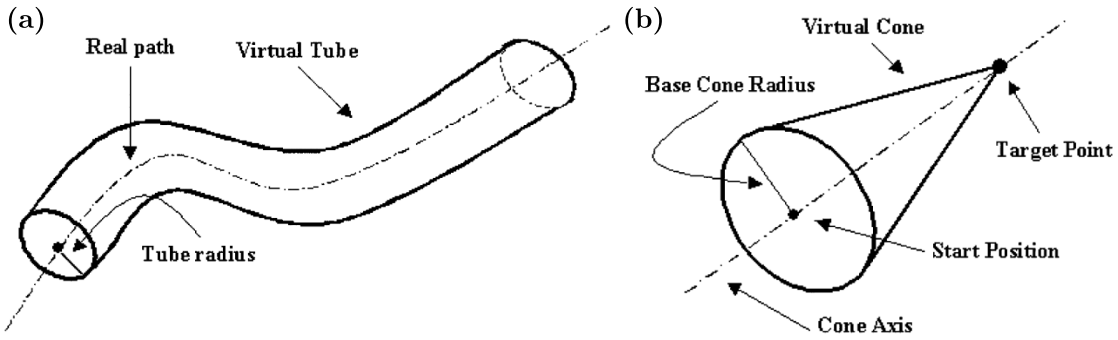

Figure 1.15: Spatial Virtual fixtures. **(a)** Virtual tube. **(b)** Virtual cone. (Reprinted with permission from Bettini et al. [67])

force and preventing entering into certain forbidden regions of the workspace. One example is the Active Constraint Robot (ACROBOT) [35], which is a small, low-powered, special purpose robot for knee surgery developed by the Imperial College London. This robot uses backdrivable motors and transmissions, so it has low mechanical impedance in each axis, allowing the robot to be moved by the surgeon with low force by pushing a handle mounted near the tip of the robot. The robot is force controlled by adjusting the torque of the motor depending on the force applied by the surgeon, and the current position of the robot in relation to the cutting boundaries. As the surgeon approaches and contacts a constraint surface defined in the preoperative plan, the relationship between the force applied by the surgeon and the torque of the motors (admittance) varies, constraining the tool motion until the edge of the permitted region, and prevents further motion outward over the boundary. The ACROBOT has been used mainly for knee surgery, but also in a range of orthopedic and soft tissue procedures.

This concept of virtual fixtures has been applied in the Johns Hopkins University Steady Hand robot system for micromanipulation [37]. It consists of a Cartesian stage allowing three orthogonal translational degrees of freedom and a Remote Center of Motion (RCM) allowing two orthogonal rotational DOFs. This robot is intended to extend the human's ability to perform small-scale manipulation tasks requiring human judgment, sensory integration, and hand-eye coordination. The tool is held simultaneously both by the surgeon's hand and the robot arm. The robot's controller senses the forces exerted by the operator on the tool and by the tool on the environment, and uses this information in various control modes to provide smooth, tremor-free precise positional control and force scaling. Applications of this robot include eye surgery, microvascular surgery and neurosurgery.

1.3.4 Robot-assisted surgical suturing

Suturing is one of the most challenging task encountered in a surgical procedure. A suturing task comprises several steps: (*s0:*) selection of suitable entry and exit points, (*s1:*) needle grasping, placement and reorientation over the entry point, (*s2:*) needle insertion and extraction, (*s3:*) create a suture loop for knot tying, and (*s4:*) tighten and secure the knot [69]. In minimally invasive surgery, suturing is a frequent, repetitive and yet time-consuming task. Previous studies have focused on the automation of one or some of the above-mentioned steps in the suturing task in order to reduce surgeon's fatigue and operation time.

Learning-by-demonstration techniques have been proposed for autonomous knot-tying tasks (*s3-s4*). Knoll et al. [72] introduced a skill transfer approach from human demonstrations based on knot-tying primitives decomposition, feature extraction, and task generalization using the da Vinci robot. However, the implementation was not suitable for online planning because of the significant computational cost. In [73], Osa et al. used a set of demonstrated trajectories under various environmental conditions to learn the knot-tying process, and proposed an online trajectory regeneration for adapting over changes of the dynamic surgical environment. On the other hand, Van der Berg et al. [74] proposed the use of iterative learning control to determine a task trajectory without the need of the task description. Different approaches have been proposed to perform autonomous surgical tasks without learning. Multi-step sequential trajectory specifically designed for knot-tying was proposed in [75]. Chow et al. [76] proposed a knot-tying automated path generation based on a binary-star search method over objective metrics defined for candidate motion patterns.

Autonomous stitching (*s0–s2*) has also been studied, mainly focused on the generation of an optimal needle trajectory, and can be divided into two types: constant curvature paths and adjustable curvature paths. In the case of a constant curvature path, the needle rotates around its center to reduce trauma when puncturing the tissue. Nageotte et al. [77] presented a kinematic analysis of the stitching task, and used an A* based method to find the optimal needle path given the desired entry and exit points. Liu et al. [78], introduced an offline optimization framework for optimal entry port selection and needle grasping pose. However, the exhaustive search methods require a high computational cost and are not suitable for real-time implementation. Staub et al. [79] proposed a visual-servoing control to position the needle over a desired point marked by a laser pointer and performs a circular needle motion to pierce the tissue. D'Ettore et al. [80] also applied a vision-based control for autonomous needle grasping (see Figure 1.16), but did not consider additional

requirements for subsequent suturing steps. Iyer et al. [81] proposed a visual-servoing control for autonomous stitching, which provides smooth needle steering by calculating the needle optimal center point of rotation. Pedram et al. [82] used a nonlinear optimization algorithm to generate a needle constant curvature path subject to the tissue geometry, desired entry/exit points, and kinematic constraints.

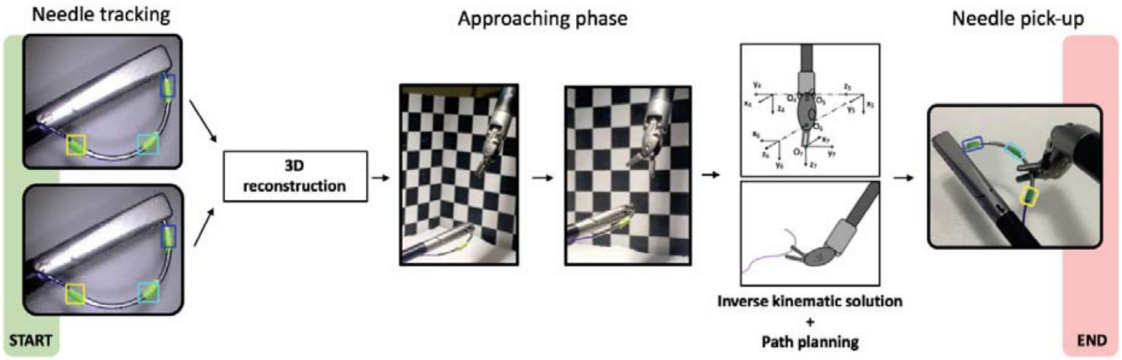

Figure 1.16: Automatick needle picking (Reprinted with permission from D'Ettore et al. [80])

Notice that the use of constant curvature paths could fail to meet the suturing requirements, e.g., depth and length, or conflict with the robot kinematic constraints. As a result, capability of exerting forces required for tissue penetration could be reduced. Therefore, subsequent studies have considered adjustable curvature paths, where needle orientation adjustments are allowed. Sen et al. [83] proposed a sequential nonconvex optimization framework to find the optimal trajectory, subject to kinematic constraints, bounded needle reorientation, minimum trajectory length and orthogonal needle poses to reduce tissue trauma (see Figure 1.17). It also includes a mechanical needle guide to reduce needle pose uncertainty. Jackson et al. [84] developed a needle trajectory plan based on the best practices of manual suturing that allows needle reorientation and ensures suture depth and needle handling. Autonomous needle extraction was proposed in [85], in which the visual feedback is used to control two teleoperated robot arms with a single user interface.

Additional studies explored the automation of other surgical sub-tasks; optimal port placement [86], surgical debridement [87], surgical cutting [88], [89] and real-time thread tracking [90],[91]. Fully autonomous suturing is still considered a high-risk procedure because of the variability in human anatomy and uncertainties in environment modeling (tissue, needle pose and thread). They also rely on time-consuming complex calibration setups that need frequent readjustments and low operational speeds that extend the oper-

ation time.

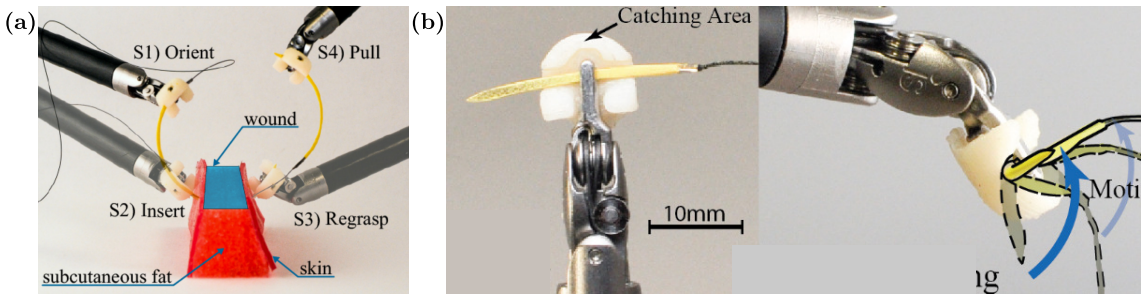

Figure 1.17: Automatic suturing proposed by Sen et al. (Reprinted with permission from Sen et al. [83])

Cooperative human-robot suturing with shared control between the operator and robot has been also explored as an alternative to autonomous suturing. Here the robot can guide or restrict the surgeon's command, or execute automated surgical subtasks. In [92], the surgeon commands the robot in subtasks where environment interactions are involved, such as grasping or needle insertion, and the automatic execution of pre-learned subtasks are executed sequentially after the manual subtasks. Reed et al. [93] developed a robot-assisted steering system for bevel-tip steerable needles by integrating a stochastic roadmap-based planer, a planar controller and a torsion compensator. During the surgical procedure, the surgeon can either only pause the insertion to verify the needle location or abort the procedure by retracting the needle.

Virtual constraints are often used to guide or constrain the surgeon's commanding motion. Kapoor et al. [69] proposed a guidance virtual fixture to assist the surgeon move towards the entry point and along the desired trajectory defined for a stitching task. The constrained motions is formulated as a quadratic programming minimization problem. The surgeon commands the robot through a force-based control, and needle reorientation is allowed to reduce the error between the desired entry/exit points. An impedance virtual fixture framework for needle passing and knot tying was introduced in [94] by constraining the tool tip within a plane and reducing needle pose uncertainty through a 3D printed needle holder. Selvaggio et al. [95] presented an optimization-based haptic shared control for needle grasping that takes into consideration the robot joint limits and singularities. Marinho et al. [96] developed a looping guidance virtual fixture and a trajectory guidance cylinder based on constrained optimization and haptic feedback to assist the surgeon during a teleoperated knot-tying task. In [97], multiple control strategies for a stitching task were compared; telemanipulation, autonomous, shared control with orientation free, and shared

control with orientation constrained.

1.4 Objective of this thesis

Robotic surgical systems have been developed to address the current manipulation and visualization limitations in minimally invasive surgery, but many challenges still remain for EES. Previous studies focused on enhancing the surgeon's capabilities during the execution of certain surgical tasks, but hindering the execution of other activities. Teleoperation systems, for example, can provide high dexterity surgical manipulation, but require a large, time-consuming setup and the help of assistance staff for tool exchange. They also result in high-cost, bulky solutions that are not suitable for highly constrained procedures such as the case of EES. The interactions between the surgeon and the robot also represent a significant issue as they occur in a dynamic and highly constrained environment, and the robotic system should be able to perceive the environment and the human intention and then adapt to both appropriately and intuitively. Thus, there is a need for surgical systems that can cover the entire EES cycle, but maintaining a dexterous, intuitive, and safe operation of the robotic system. This work aims to fill this gap by presenting a cooperative framework for a robotic-assisted surgical system targeting EES applications. For this thesis, the term cooperative is understood as the robot capability to cooperate in a surgical procedure by guiding or restricting the surgeon's command, and executing surgical subtasks autonomously under the surgeon's supervision.

The original contributions of this thesis can be summarized as follows:

- Conceptualization, design, and implementation of a multi-level cooperative framework for robotic-assisted EES.
- Design and integration of a new, compact, and versatile hybrid human-robot interface.
- Stitching task characterization for EES and semiautonomous functionality.
- Validation experiments for the proposed surgical system in a EES phantom environment.

We proposed a new, compact, and versatile human-robot interface. We describe the mechanical design, control strategy, and system integration for robotic-assisted endonasal surgery. The interface design considers specific EES challenges, providing an easy setup and tool exchange, ergonomic operation, and fast access to the patient, which allow the

surgeon to use additional tools within the surgical workspace. The proposed system follows a hybrid approach, where a force-controlled interface and a serial-link interface are compactly embedded in a module attached to each robot arm. We define a control strategy in a multistate simplified surgical task scenario, with four basic instrument operations such as positioning, insertion, manipulation, and extraction. The safety requirements are addressed through a combination of virtual remote-center-of-motion (VRCM), workspace virtual fixtures, and force-based control, defined for each operation state.

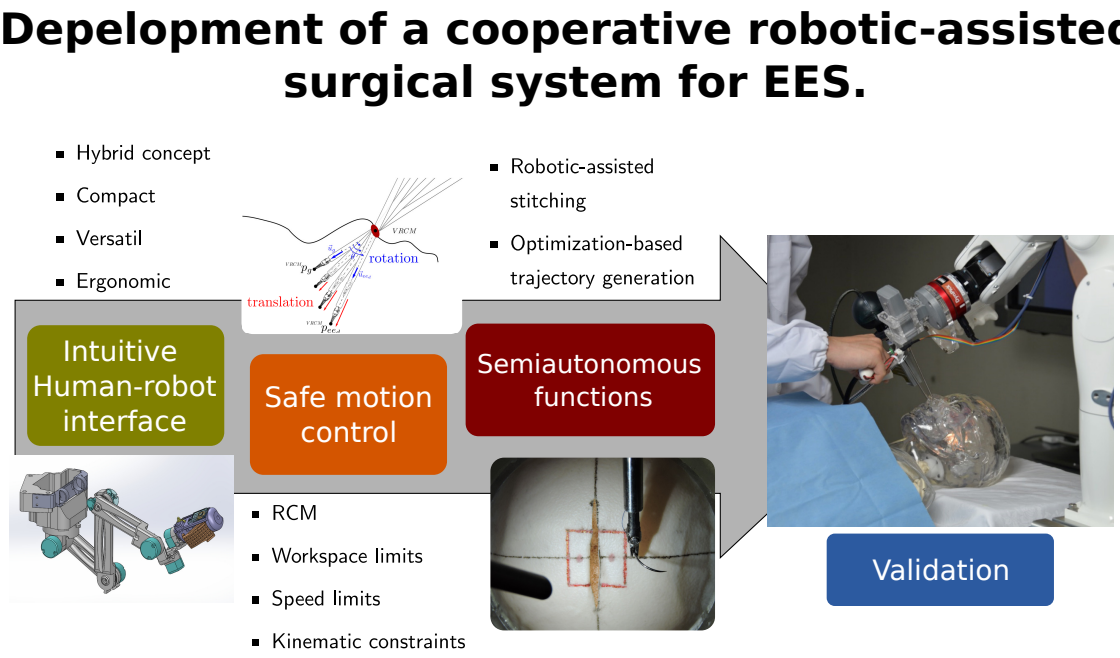

Figure 1.18: Thesis objectives

We also introduced the developing of a robot-assisted stitching framework based on optimal needle trajectory generation subject to stitching kinematic constraints, and a dual concurrent constrained control algorithm to track the generated trajectory. The proposed approach restrain and guides the motion of the robot during needle insertion and extraction through the tissue, while preserving a remote center of motion (RCM) placed over the nostril to reduce additional tissue damage. We demonstrate the effectiveness of the proposed system by reachability, object manipulation, and surgical dexterity tasks in a phantom environment designed to reproduce the constraints found in a real scenario.

1.5 Thesis organization

The outline of this thesis is summarized in Fig. 1.19. Chapter 1 introduces the current status, challenges, and existing problems in endoscopic endonasal surgery. This chapter provides a classification and brief description of the most relevant robotic systems used for surgical applications. This chapter also summarizes related works in the development of robotic-assisted surgical systems for EES, cooperative control, spatial motion constraints and robot-assisted suturing. It describes the benefits of robotic technology in surgery and leads to the importance of this work.

In chapter 2, we introduce the architecture of our proposed cooperative robotic-assisted surgical system for EES. We describe the software and hardware implementations, components, control strategy and system integration. The system is proposed to provide multiple levels of assistance depending of the subtask to perform.

Chapter 3 focuses on the human-robot interface design. We proposed a hybrid interface comprised of an force-based interface and a serial-link interface. In this chapter we describe the design considerations, the evolution in the mechanical design and the kinematic analysis.

Chapter 4 describes the cooperative approach for the robot motion control and virtual constraints implemented to ensure safe robot motion. It presents the control strategies developed for each surgical task stage and the experimental evaluation of the proposed method.

In Chapter 5, we present an online trajectory generation for a stitching task with circular needles. It describes the characterization of the stitching task and introduces the nonlinear optimization framework used to generate a constrained trajectory subject to the task kinematic constraints. Through simulation and physical experiments, we validate the performance of the proposed method.

Finally, chapter 6 summarizes the contributions of the proposed system, describes the shortcomings of the current work, and finally discusses the future research direction.

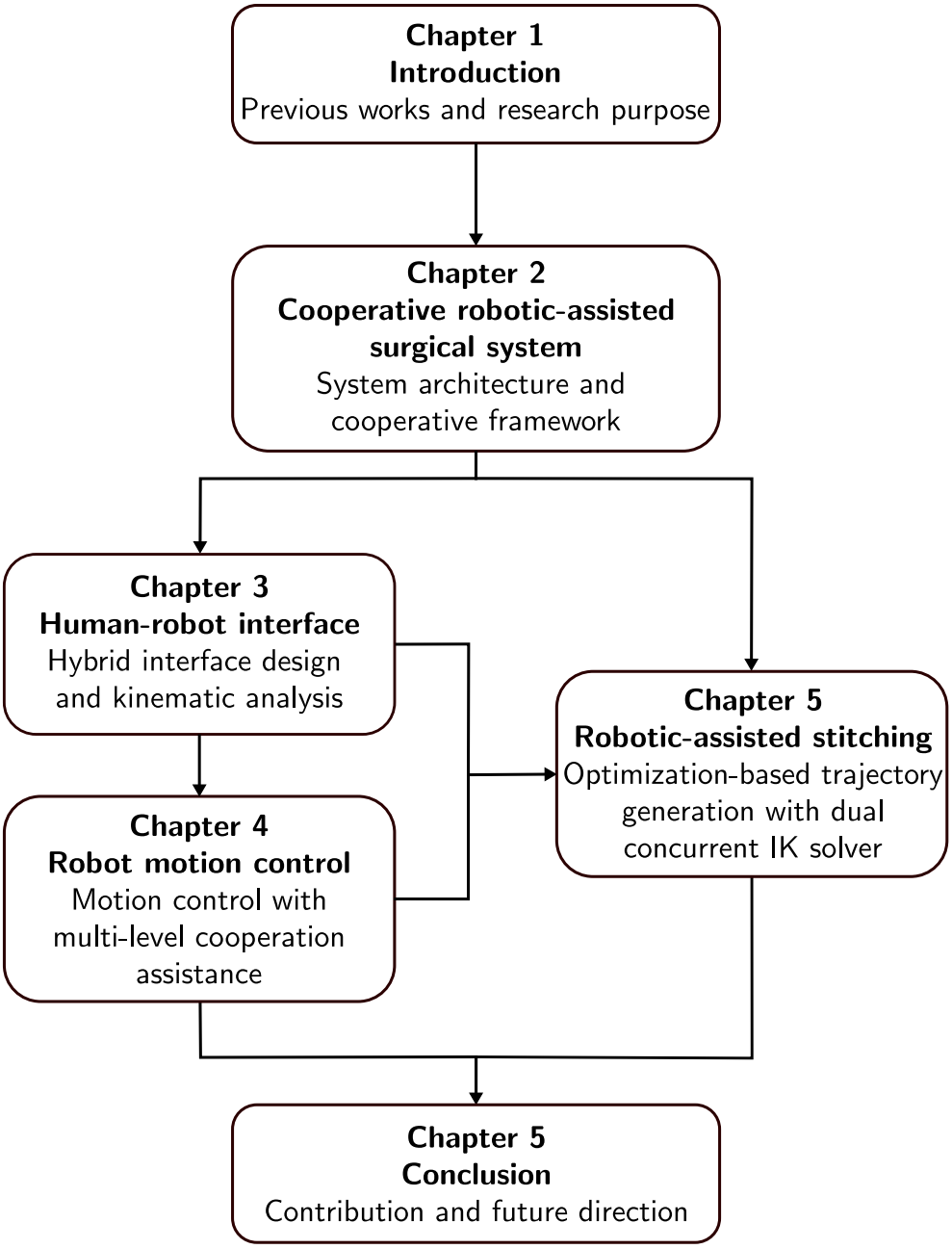

Figure 1.19: Thesis organization.

Chapter 2

Cooperative surgical system

Chapter 1 introduced an overview of relevant previous works in the development of robotic-assisted surgical systems, with an emphasis on cooperative frameworks. This chapter describes the design considerations for a minimally invasive endoscopic endonasal approach and the implementation of a cooperative framework for EES. Section 2.1 gives an discusses the design requirements defined for an EES application. Using this requirements, an overview of the proposed robotic surgical system is provided in Section 2.2, including the software architecture and system integration. Section 2.3 describes the multi-level cooperative scheme designed to satisfy the assistance requirements on each EES subtask.

2.1 Surgical system design requirements

2.1.1 Endonasal surgery workspace

Endonasal surgery typically targets pituitary adenomas and related lesions such as Rathke's cleft cysts. EES for sellar and parasellar tumors follows a mononostril or binostril technique with the instrument trajectory starting at the nostrils through the nasal cavity into the sphenoid sinus. Once the sphenoid sinus has been reached, the sellar face bone is removed to expose the underlying dura. We characterized the EES workspace by using a 3D design of a human nasal model (M01-SET-TSPS-E1, SurgTrainer Ltd., Ibaraki, Japan), as shown in Figure 2.1. The instrument trajectory was enclosed within channels represented with two truncated cones originated from each nostril to the sella region, as illustrated in Figure 2.2b. Each nostril orifice was approximated as an ellipse with the dimensions of 15 mm $\times$ 30 mm. The sella region was represented as a sphere with a radius of 8 mm at the other end of the cones. In this model, the length of the cone was approximately 100 mm.

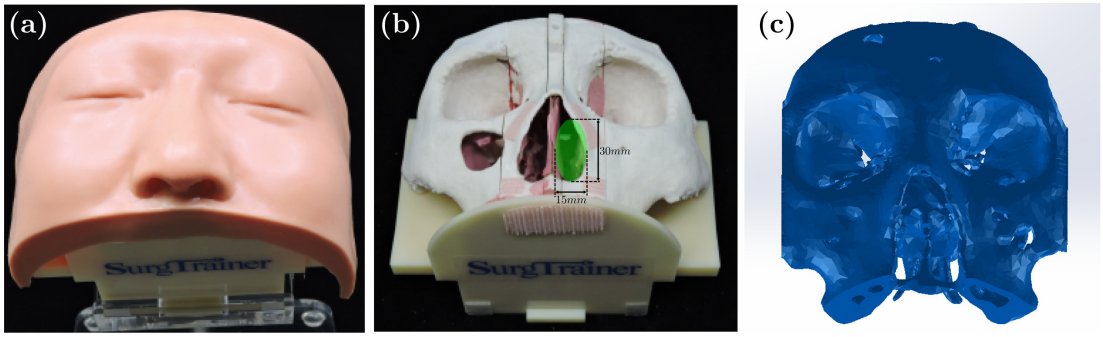

Figure 2.1: (**a**,**b**) Physical human nasal model for endonasal surgery training. (**c**) 3D model of a human nasal model.

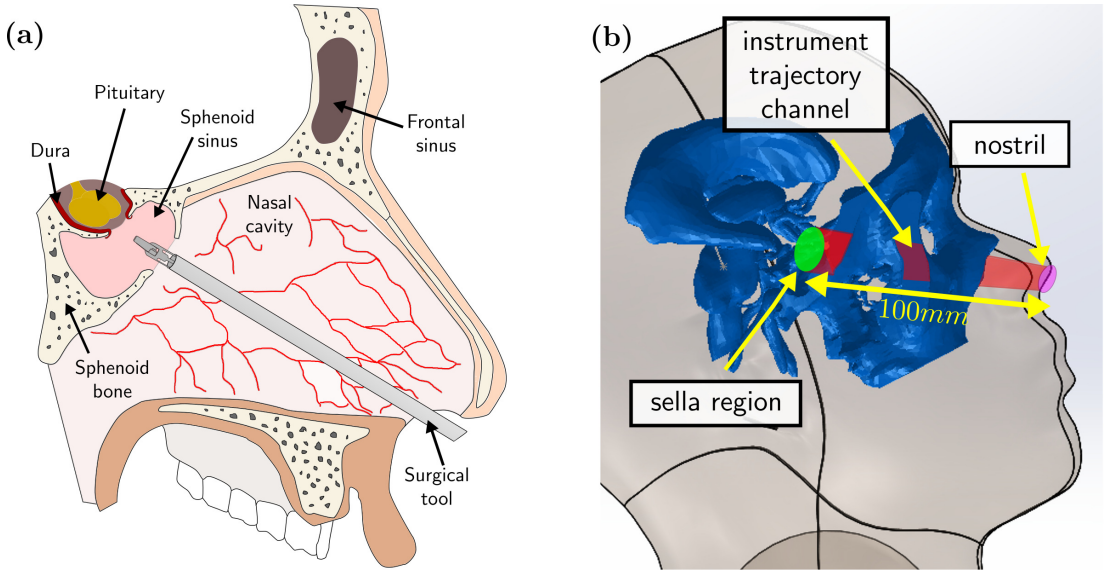

Figure 2.2: (**a**) Endoscopic endonasal surgery. (**b**) Instrument trajectory channel and the sellar target representation.

2.1.2 Robotic system requirements

The proposed robotic-assisted surgical system should meet the following requirements identified for EES:

- Arm robot: High structural stiffness, low friction, backlash, and inertia to ensure high transparency in force reflection. Portable and compact for quick preoperational setup.

- Surgical instrument shape and weight: Light weight for easy and fast instrument exchange. Reduced size is needed to avoid collision or obstruction of additional instruments or equipment in a constrained workspace.

- Greater number of degrees of freedom: In EES, the surgical instrument pivots at the nasal port, so the manipulation is highly constrained. The surgical instrument motion is reduced to only four degrees of freedom, i.e., three rotations about the incision point (Pitch, yaw, and roll) and one translational motion along the direction of insertion of the instrument. Therefore, appropriate and sufficient DOFs have to be provided.

- Workspace: Defined by the anatomical constraints in the nasal cavity and sphenoid sinus (see Figure 2.2).

- Mirror effect: The movements of the surgeon's hand and the instrument tip are in opposite directions because of its rotation around a fixed point. It is highly desired that the robotic system eliminates this effect and allows the surgeons to project their hand's motion into the robotic instruments inside the patient's body.

- Accuracy: Accuracies of $\leq 2mm$ are considered acceptable for neurosurgical applications.

- Visualization: Although 3D endoscope technology exists, the use of 2D endoscopes is still common. The system should be fully functional under a common 2D visualization.

- Instrument exchange: Exchange of surgical instruments is a frequent task for EES. Reduction of the exchange time or frequency is highly desirable.

- Autonomous functions: To reduce the stress of frequent task such as tissue stitching.

- Safety: High degree of security and reliability. Position and velocity constraints should be imposed over the surgical instrument. Real-time operation monitoring and multi-layer security measures should be implemented.

- Ergonomics: EES procedures last between two to four hours. The surgeon should be able to operate the robotic system in a comfortable and relaxed posture.

- Intuitive operation: The userinterface should provide intuitive manipulation for highly dexterous tasks such as suturing.

The quantitative requirements are summarized in Table 2.1.

Table 2.1: Surgical Requirements and System Design

Parameter		Value
Accuracy		$\leq 2mm$
Surgical instrument DOFs		≥ 2
Maximum velocity	Insertion	0.2m/s
	Lateral	0.2m/s
	Roll	30rpm

2.2 Robotic Surgical System

2.2.1 Robotic surgical system overview

The proposed robotic surgical system is based on the SmartArm concept [98] (see Figure 2.3) composed of two sets of an industrial robot arm, articulated forceps, and the proposed user interface.

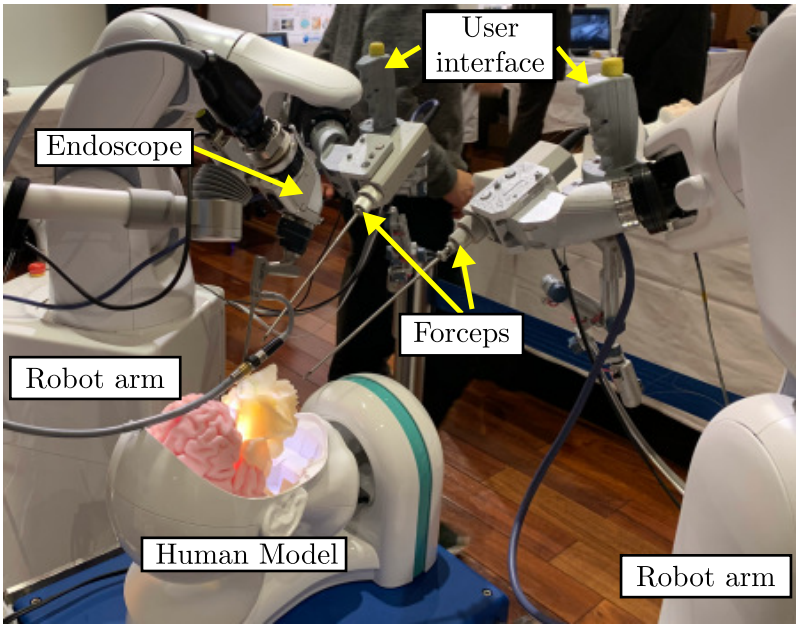

Figure 2.3: The proposed robotic surgical system based on the SmartArm concept for endoscopic endonasal surgery (EES).

Industrial robot arm

Two 6-DOF industrial robot arms (VS-050, DENSO Corporation, Aichi, Japan) are used. The VS-050 robot arm provides a maximum motion area of 505mm and a maximum payload of 4kg. Each arm is controlled by an industrial controller (RC8, DENSO Corporation, Aichi, Japan), which allows end-effector Cartesian and joint position control.

User interface

The user interface is attached to the robot arm and holds a multi-DOF articulated forceps. It is designed to provide intuitive and simultaneous control of multiple DOFs of the robot arm and the articulated forceps. The concept design and kinematic analysis are described in Chapter 3.

Multi-DOF articulated forceps

In this work, we used two types of articulated forceps: a 2-DOF forceps built in our laboratory and a 4-DOF forceps developed in [99]. The 4-DOF articulated forceps (see Figure 2.4a) comprises a shaft (diameter: 3.5 mm and length: 233 mm), a triangular-shaped gripper (length: 4 mm), and elastic elements to provide 3-DOF tip movement (bending in two directions and rolling around the axis) and the grasping function. The 4-DOF forceps tip motion is controlled by five DC motors (four motors for bending and grasping and one motor for the roll motion). The 2-DOF forceps (see Figure 2.4b) was built in our laboratory by modifying disposable flexible biopsy forceps (FB-231D, Olympus, Tokyo, Japan) and can be driven by the 4-DOF forceps actuator unit. The 2-DOF forceps comprises a shaft (diameter: 2 mm and length: 245 mm) and a round-shaped (cupped) gripper (length: 3 mm) and provides the roll and grasping functions. A simplified inverse kinematics used to control the forceps tip motion is defined as:

$$d_1 = -t * \theta + r * \psi - \alpha - K_1|\theta| \tag{2.1}$$

$$d_2 = -R * \theta \tag{2.2}$$

$$d_1 = t * \theta - r * \psi - \alpha - K_1|\theta| \tag{2.3}$$

$$d_4 = R * \theta \tag{2.4}$$

$$d_5 = \phi \tag{2.5}$$

where:

d_1 : Spindle Motor 1 position (mm)

d_2 : Spindle Motor 2 position (mm)
d_3 : Spindle Motor 3 position (mm)
d_4 : Spindle Motor 4 position (mm)
d_5 : Rotational Motor position (rad)
ψ : Yaw bending angle (rad)
θ : Pitch bending angle (rad)
ϕ : Roll angle (rad)
α : Initial opening angle offset (rad)
K_1 : Offset coefficient (rad)

The offset corresponding to $\alpha - K_1|\theta|$ has been obtained experimentally.

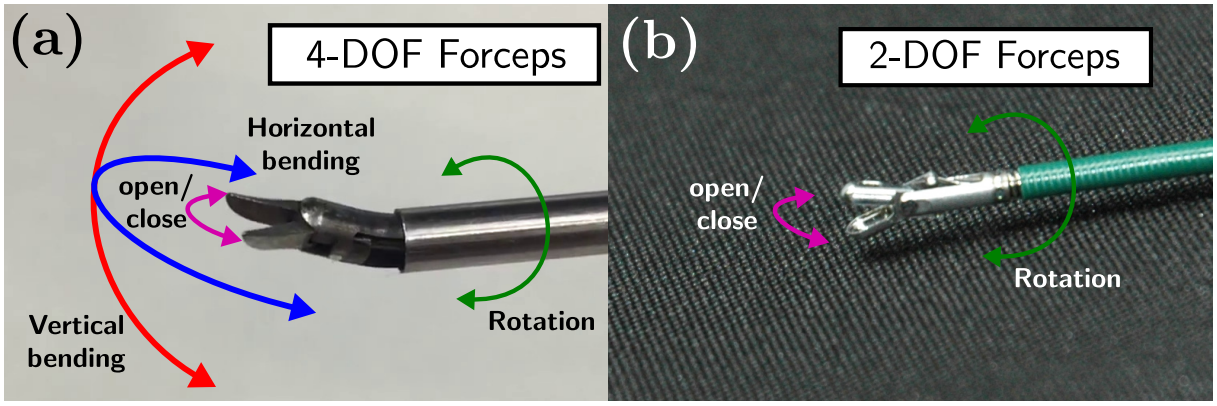

Figure 2.4: (**a**) 4-DOF articulated forceps. (**b**) 2-DOF articulated forceps.

2.2.2 Robotic frames description

In the proposed system, each robotic unit has $6+n$-DOF, where 6-DOF were provided by the robot arm and $n = 2$- or 4-DOF are the contribution from the multi-DOF articulated forceps. The coordinate frames for the robotic system are defined as shown in Figure 2.5, where F_B represents the base frame, F_a is the robot arm tool frame, F_i is the interface's base frame, F_h is the interface's gripper for the surgeon's hand, F_{ee} is the forceps tip frame, F_{rcm} is the RCM frame at the nostril, and F_g corresponds to the target frame in the sella region. Relative coordinate frames are defined as ${}^aR_b$ where b refers to the local frame and a refers to the reference frame. For notational convenience, a full pose (including position and orientation) in the frame a is denoted by ${}^aX$ as a combination of a position vector ${}^ap$ and a rotation matrix ${}^aR$. When no reference frame is specified, it is assumed to be expressed in the base frame F_b.

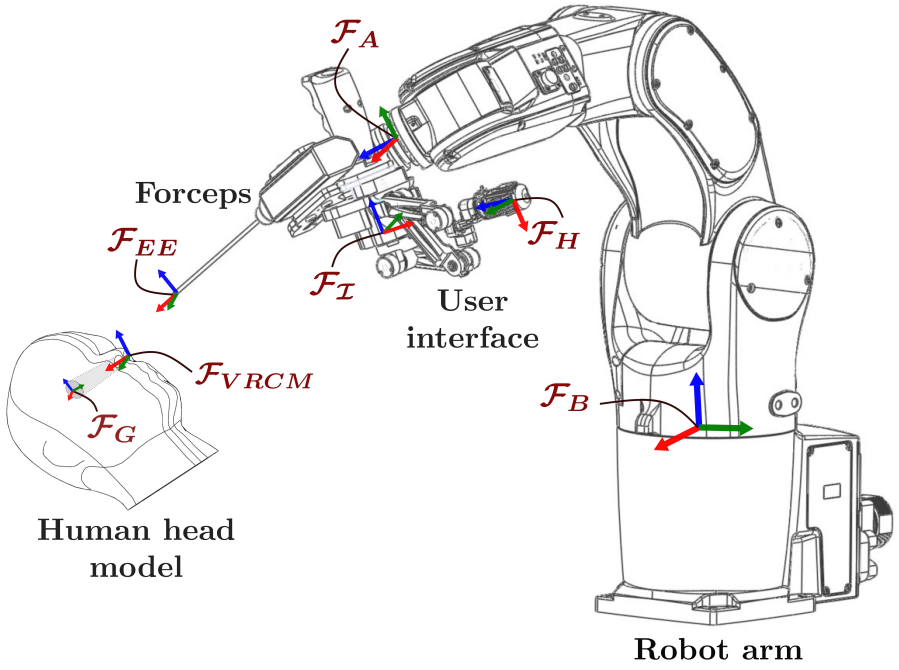

Figure 2.5: Definition of the coordinate frames on the proposed robotic surgical system.

2.2.3 Software architecture

The developed system architecture is depicted in Figure 2.6. In the proposed robotic surgical system, the control scheme consisted of two levels: one was a low-level joint motion controller, and the other was a high-level robot controller. At the low level, an industrial robot controller (RC8, DENSO Corporation, Aichi, Japan) controls the manipulator joint positions, monitors arm singularities, and limits the joint velocities and workspace for safety motion. Motor drivers (ESCON, Maxon Motor, Sachseln, Switzerland) in the forceps controller box control the multi-DOF forceps joint position. The high-level controller provides the target robot pose based on the input from the interface devices.

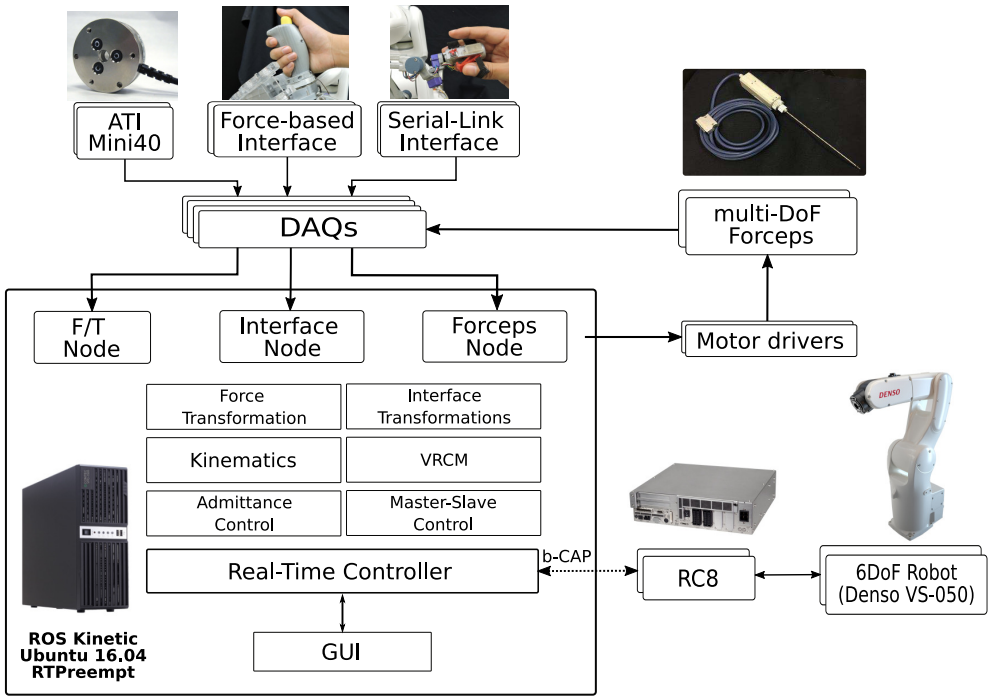

Figure 2.6: Software architecture of the proposed robotic surgical system.

The proposed control system is implemented on a 2.4 GHz Core i7 CONTEC computer running Linux (Ubuntu 16.04, Canonical) with real-time patches (RT-PREEMPT) and the Robot Operating System (ROS) framework on top of it. For the communication between the high-level controller and the robot manipulator, the b-CAP protocol is implemented with the C++ libraries provided by the robot manufacturer. The user interface devices are connected to 16-bit data acquisition (DAQ) systems (PCI-1227U-AE, Advantech, USA), and the forceps low-level controller is connected to analog input/output boards (PCI-1216U-AE, Advantech, USA) and four-axis encoder boards (PCI-1284-AE, Advantech, USA). The control loop runs at a rate of 500 Hz in synchronous mode with the RC8. In each cycle, the current robot pose is updated, and then, the target robot pose is computed and sent to the RC8 and the forceps controller box. In order to provide a modular and scalable architecture, communication with the interface devices was implemented as ROS nodes in the middleware level.

The middleware is implemented as ROS nodes for the communications with the interface devices and to provide a modular and scalable architecture. The different modules implemented are:

- GUI: Provides an interface for the user to select the desired behavior for the robot

system written in C++ and the Qt5 framework. It facilitates the control of the arm robots, user interface, and multi-DOF forceps (see Figure 2.7);

- Force sensing: Updates the transformed force/torque readings from the force sensors at a rate of 1KHz;
- Interface command: Provides the desired input command obtained from the hybrid user interface at each cooperation state;
- Admittance Controller: Implements an admittance controller in Cartesian space;
- Teleoperation Controller: Implements a teleoperation controller in Cartesian space;
- Kinematics: Implements algorithms for the computation of robot forward kinematics and inverse kinematics;
- VRCM: Implements a virtual remote center of motion to constrain the forceps motion
- Forceps Control: Position controller of the articulated forceps DOFs;

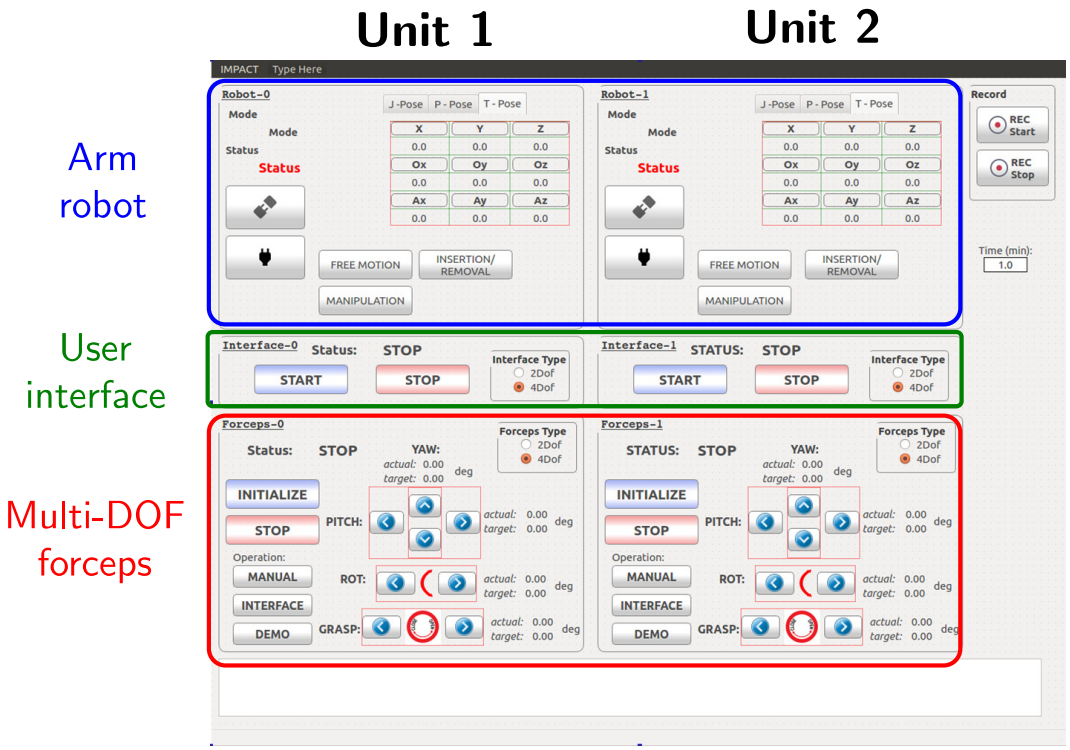

Figure 2.7: Graphical User Interface for robotic system control.

2.3 Cooperative scheme for an endoscopic endonasal surgical task

We identified the following four repetitive activities during a common endonasal surgical operation: (1) positioning, (2) insertion, (3) manipulation, and (4) extraction. The surgeon first positions the tip of the surgical instrument close to the nostril orifice and inserts the surgical instrument through the nasal cavity constrained by the nostril. Subsequently, once the target was reached, the surgeon performs a manipulation task. Finally, the instrument is extracted to replace the tool or finish the surgical procedure. Each state imposes its assistance requirements.

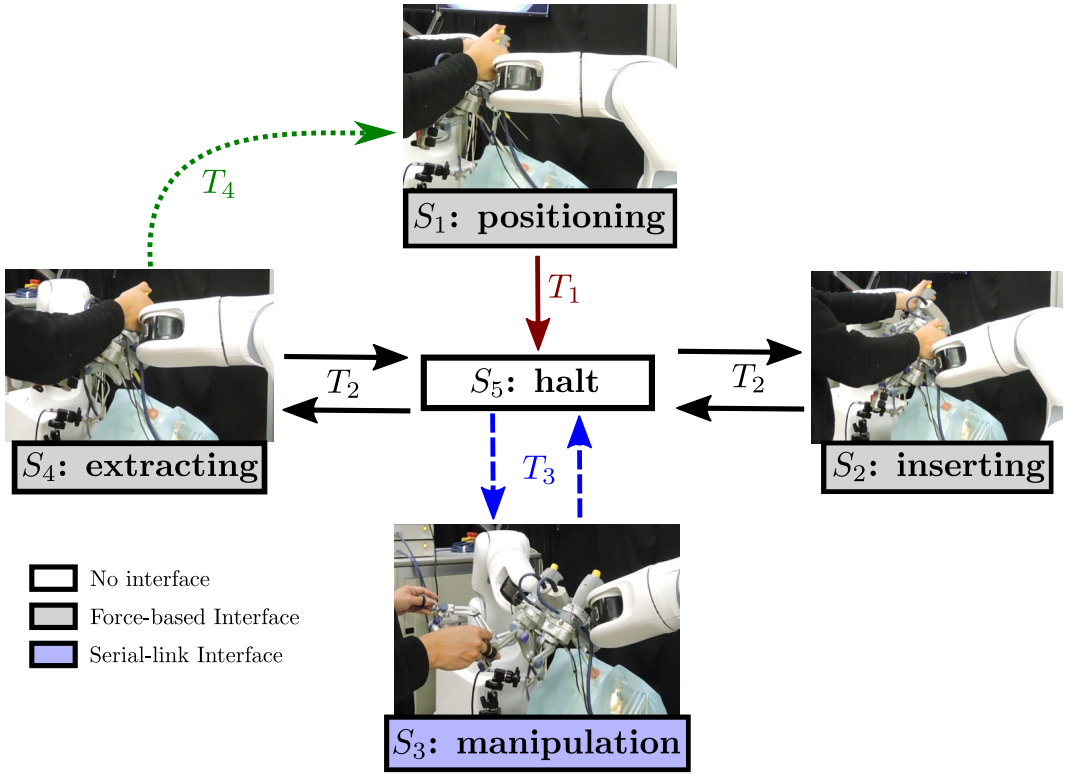

Figure 2.8: State machine for surgical task characterization. The states defined are S_1: positioning, S_2: insertion, S_3: manipulation, S_4: extracting, and S_5: halt. Transition states are represented with arrows: T_1 and T_2 are activated by the handle push-button; T_3 is activated by the gripper push-button; and T_4 is automatically activated when the tool is outside of the nostril. The virtual remote-center-of-motion (VRCM) position is set at the beginning of T_1.

We propose a cooperative scheme designed to satisfy all these requirements at all these stages by using a user interface composed of a force-based interface and a remote-controlled serial-link interface attached in a single device to control the position and orientation of the articulated forceps. Different levels of assistance were provided by a state machine implemented with six states S_n ($n \in \{1, ..., 6\}$) depending on the current surgical task, as shown in Figure 2.8, where S_1 is positioning, S_2 is insertion, S_3 is manipulation, S_4 is extraction, S_5 is halt, and S_6 is suture.

Positioning (S_1)

First, the surgeon positions the forceps from a neutral position to the nostril entrance using the force-based interface.

Insertion (S_2)

Then, a virtual remote-center-of-motion (RCM) is set at this location, which constrained the motion of the forceps during the subsequent insertion sequence .

Manipulation (S_3)

Once having reached the target zone, in the manipulation phase, the serial-link interface was used to control the articulated forceps tip motion and the robotic arm pose, simultaneously preserving the VRCM constraints.

Extraction (S_4)

Finally, the forceps were extracted again using the force-based interface. Once outside the nose, the RCM is deactivated, and the robot can be freely positioned to a neutral position by the force-based interface.

Halt (S_5)

Halting the robot is allowed at any of these stages.

Suturing (S_6)

Suturing state can only be reached from a manipulation state (S_3). It provides guidance virtual fixtures and semi-autonomous motion for specific suturing tasks (e.g., stitching).

As described above, the use of the interface is switched according to each state. The force-based interface is used during S_1 (positioning), S_2 (insertion), and S_4 (extraction); and the remote-controlled interface was used during S_3 (manipulation), and S_6 (suturing).

Chapter 3

Human-Robot interface

This chapter describes the design and functioning of a human-robot user interface developed to satisfy the requirements of a robotic-assisted EES. Section 3.1 discusses multiple user interface design concepts developed to satisfy the cooperative framework. It provides insights into the advantages and challenges faced with each interface type. Section 3.2 describes in more detail the hybrid concept of the user interface selected for the proposed cooperative surgical system.

3.1 Development of interface design

The user-interface design followed an iterative process of continuous prototyping and evaluation of multiple designs until all the requirements established in Section 2.1.2 were satisfied. The design requirements can be summarized as follows:

- Bimanual intuitive operation
- Simultaneous control of 8–10 DOFs (arm robot:6 DOF, articulated forceps: 2–4 DOF)
- Ergonomic
- Placed in the surgical workspace
- Lightweight, compact and portable

The main challenge in the design of a user interface for EES is to find an intuitive control of multiple degrees of freedom (six DOFs from the arm robot and two/four DOFs from the articulated forceps). The user interface has to be attached to the robot arm. In this way, the surgeon has direct access to the patient and can manipulate additional

instruments (e.g. endoscope camera). We developed and evaluated multiple user interface design concepts to find a suitable balance between intuitiveness, ergonomic, and dexterity. The following five types of user interface design were studied:

1. 2-DOF Gripper Interface
2. Dual force sensor Interface
3. Joystick-based Interface
4. 6-DOF serial-link Interface
5. 7-DOF serial-link Interface

The main advantages and difficulties encountered for each user interface type will be summarized in the following sections.

3.1.1 2-DOF Gripper interfaces

The first interface developed consists of a handle with a gripper interface attached to the forceps holder as shown in Figure 3.1.

The thumb and index fingers are placed in the gripper interface and secured with Velcro straps. The gripper open/close is designed to be virtually connected to the open/close motion of the forceps jaw. Rotation along the gripper axis is also possible. Internal torsion springs keep the gripper in a base open position and provide force feedback proportionally to the gripper angle. A force sensor is placed above the forceps holder to control the arm robot's motion. The forces applied over the interface handle are used as input for a force-based controller.

The 2-DOF gripper interface is simple to develop and provides intuitive control for the open/close and rotation of the forceps jaw. However, it is not possible to control a 4-DOF forceps. Moreover, the forces exerted over the interface handle are attenuated and absorbed by the mechanical structure and are not completely transmitted to the force sensor. This makes the force control difficult to operate. Non-desired movements in the arm robot were also produced by unintentional forces generated when manipulating the gripper. The user needs to be aware of the forces applied over the handle, which decreases the interface transparency.

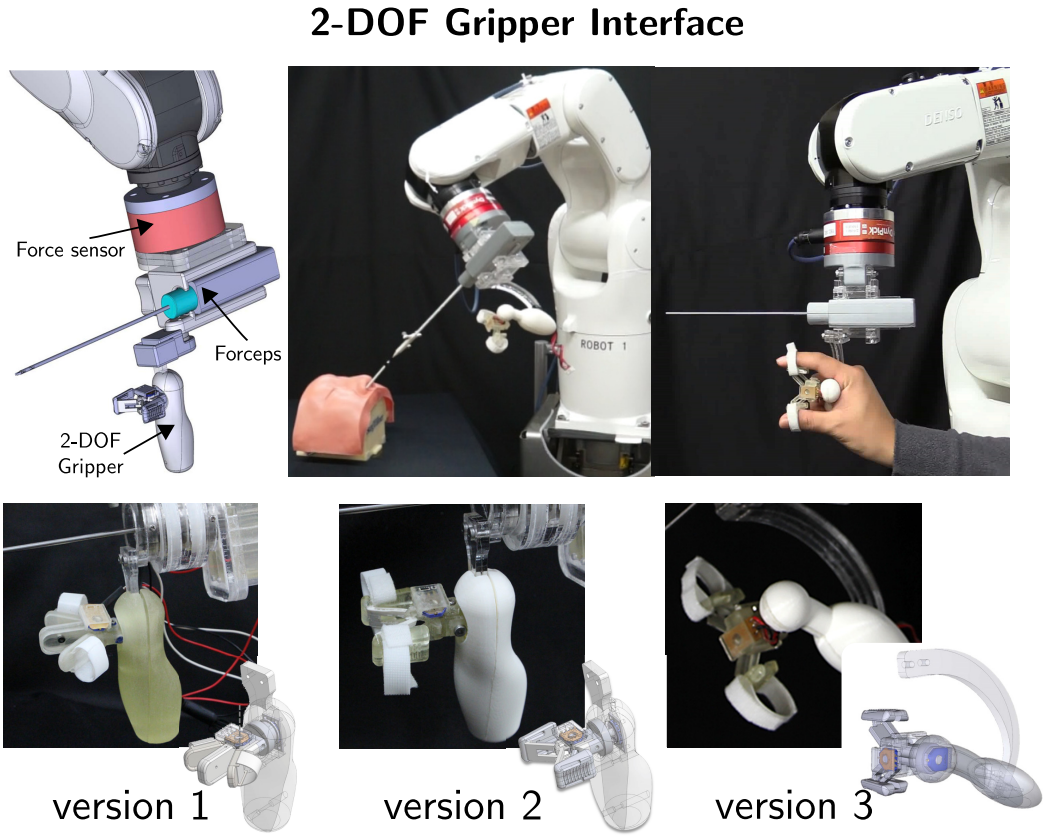

Figure 3.1: 2-DOF gripper interfaces. Interface version 1 and version 2 used a vertical handle. Version 2 modified the gripper to keep the finger parallel to the gripper axis. Version 3 used a horizontal handle to enhance hand ergonomic.

3.1.2 Dual force sensor interfaces

To be able to control the 4-DOF forceps, the interface has to be able to provide additional information about the user intention (more than 2-DOFs). We developed a dual force sensor interface that consists of two handles, one attached to the robot flange and the other supported by the forceps holder as shown in Figure 3.2. The second holder includes a 2-DOF gripper similar to the one presented in 3.1.1. Two force/torque sensors are used to receive the forces applied over each handle. The upper force sensor is used to control only the arm robot motion, whereas the second force sensor is used to control the robot and the forceps simultaneously. The main advantage of this interface is that avoids the force transmission problems of the 2-DOF gripper interface by placing a force sensor in direct contact with the holder. However, as the input forces have to be in the opposite direction of the desired robot motion, the operation is counterintuitive and hinders the

gripper manipulation. Additionally, this interface also can not be used to control the 4-DOF forceps and the additional cost of a second force sensor is high.

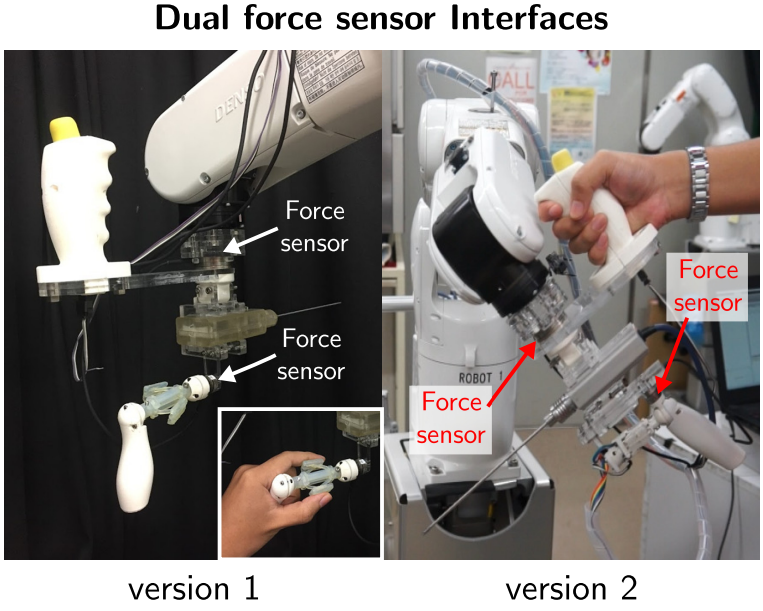

Figure 3.2: Dual force sensor interfaces. Two sensor are used to control the robot arm and the multi-DOF forceps. Version 1 placed the the additional force sensor in front of the interface gripper. Version 2 placed the additional force sensor above the interface handle.

3.1.3 Joystick-based interfaces

The previous interfaces used the gripper angles as input targets to control the position of the forceps, and the forces applied to the interface to control the robot arm velocity. In the case of the joystick-based interfaces, we used switches and buttons to change the robot velocity and forceps position discretely. Figure 3.3 shows the two versions developed for this type of interface. For version 1, we used an Oculus Touch, a hand controller that provides haptic tracking for virtual reality. We placed the Oculus Touch below the forceps holder. The Oculus Touch lever switch and buttons were used to control the forceps, while the Joystick is used to control the arm robot's motion. To further increase intuitiveness, we prototyped a 3D printed version with a dial instead of buttons to control the forceps bending angle and a soft material cover for increased ergonomics. The main difficulty for this interface was to define the velocity change steps for a smooth motion. Additionally, the use of simultaneous combinations of multiple switches and buttons increased the user's stress load.

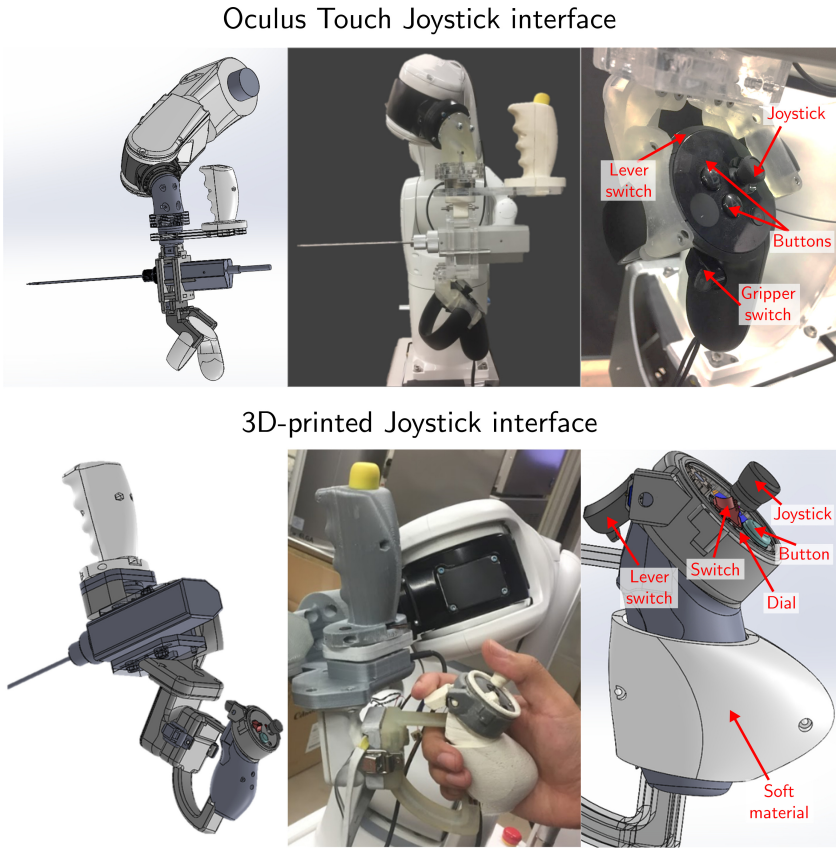

Figure 3.3: Joystick-based interfaces. The upper interface uses an Oculus Touch to generate the input commands. In the image below presents a 3D printed joystick interface based on the Oculus Touch but designed considering the specific requirements of the robotic-assisted EES.

3.1.4 Serial-link interfaces

In 2.3 we presented a state machine to characterize a surgical task. In the serial-link interface, we first use a force-based interface to control the arm robot during coarse motions (e.g., tool positioning). To provide precise and intuitive control of the multi-DOF forceps, we then proposed the use of a serial-link interface as shown in Figure 3.4. The interface does not directly split the input command between the robot and forceps but generates the desired forceps pose (position and orientation), and the robot motion controller is the one in charge to split the signal command for the robot and the forceps. This interface does not suffer from the problems of using a force signal (force attenuation) and eliminates the cross-motion presented in previous interface designs, which increases the operation intuitiveness.

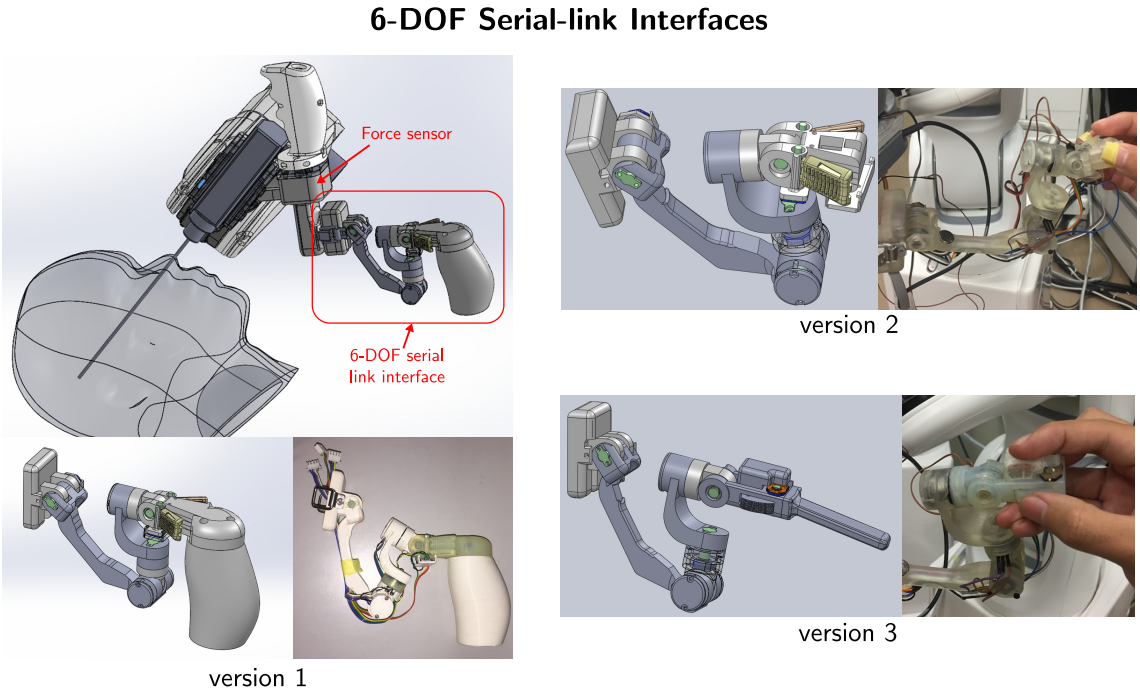

Figure 3.4: 6-DOF serial link interfaces. The difference between the three versions developed are the link lengths and the presence of a handle. Version 1 includes a vertical handle, whereas version 2 only used finger loops to secure the fingers. Version 3 included a removable pen-type handle and a switch to transition between multiple robot modes.

The workspace limitation represents the main issue in this type of interface. As the robot moves, the interface reference frame also moves. To avoid reaching the interface workspace limits, we include an additional degree of freedom as shown in Figure 3.5. A workspace and kinematics analysis of this interface design concept is presented in Section 3.2.

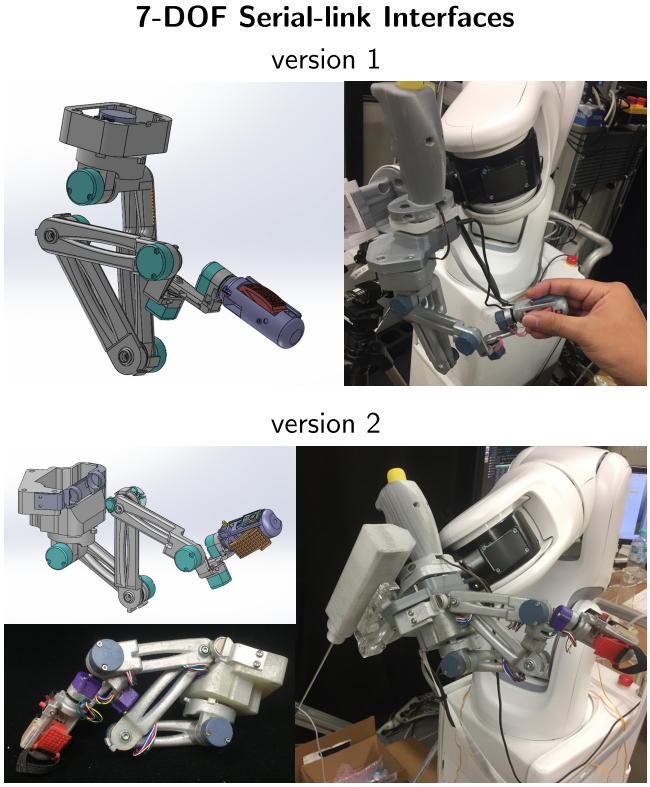

Figure 3.5: 7-DOF serial link interfaces. Version 2 includes a modification in the joint limits and link lengths to avoid mechanical singularities when two links are in parallel.

3.2 Interface description and kinematic analysis

The proposed interface (see Figure 3.6 a–c) was equipped with an ergonomic vertical handle attached to a 6 axis force/torque sensor (Mini40, ATI Industrial Automation, North Carolina, USA), with specifications described in Table 3.1, and a serial-link mechanism placed below. This interface also included a forceps slot for quick instrument exchange.

Table 3.1: Specifications for the 6-axis ATI Mini40 Force/Torque Sensor

Specification	Value
Force sensing range (x,y,z-axis)	$\pm 40N/120N$
Torque sensing range (x,y,z-axis)	$\pm 1Nm$
Weight	39 g
Size	⌀40mm x 14mm

In the force-based control, the force exerted over the handle is used to operate the robot arm. The serial-link mechanism is composed of seven revolute joints and a gripper with finger loops for the middle finger and thumb, as well as a push-button on the handle. This serial-link interface provides a 6-DOF range of motion, allowing an additional redundant DOF to increase mobility and facilitate decoupling between the articulated interface motion and the robotic arm motion, i.e., the robot arm velocity ${}^{B}\dot{p}_{A}$ does not produce a change in the position of the interface's gripper ${}^{B}\dot{p}_{H} = 0$. This can be achieved by compensating the robot arm motion with an internal motion of the interface joints. The gripper controls the open/close motion of the forceps tip, and the push-button near the gripper activates the resting mode that kept the position of the robot fixed, e.g., while the surgeon was at rest or manipulating additional instruments. Each joint of the serial-link mechanism had a rotary position sensor (SV01, Murata) to measure the angle of the joint.

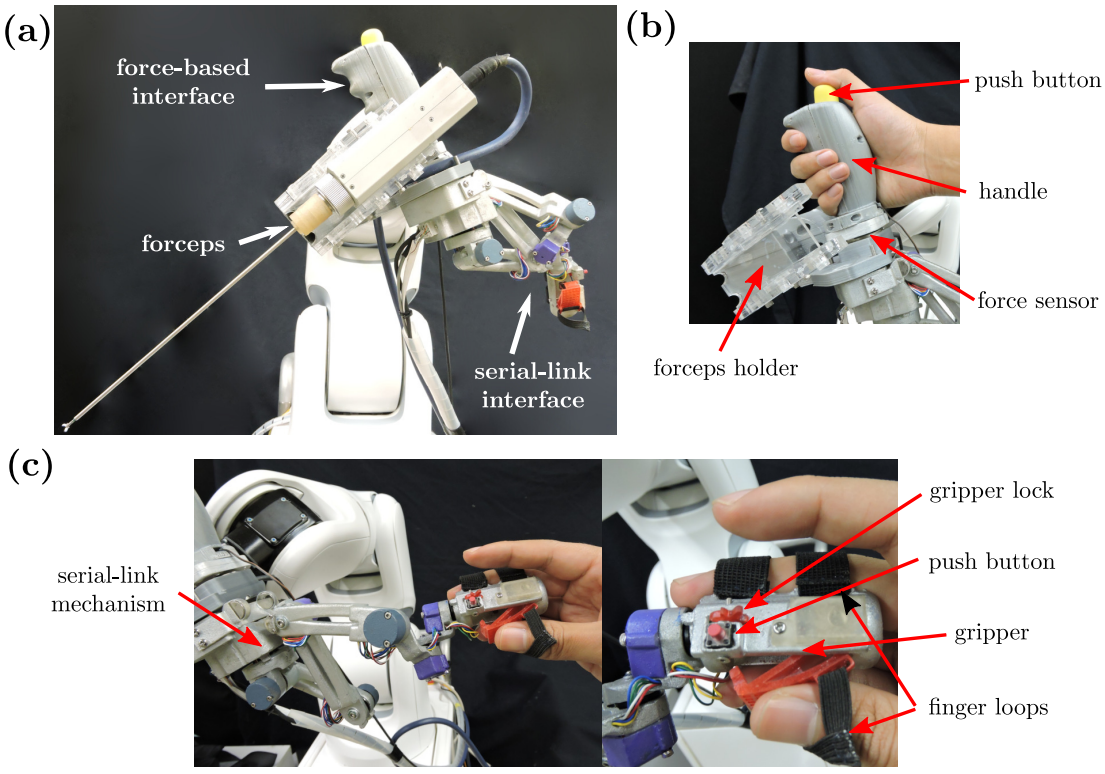

Figure 3.6: (**a**) The proposed human-robot interface. (**b**) Handle attached to a force sensor for the force-controlled interface. (**c**) Serial-link mechanism for the remote-controlled interface.

To provide a comfortable position of the surgeon's hands, the neutral position of the forceps tip is mapped with a 45° offset with respect to the wrist axis, as depicted in Figure 3.7. The workspace of the serial-link mechanism is shown in Figure 3.8, and the

detailed kinematic structure with the D-H parameters is depicted in Figure 3.9.

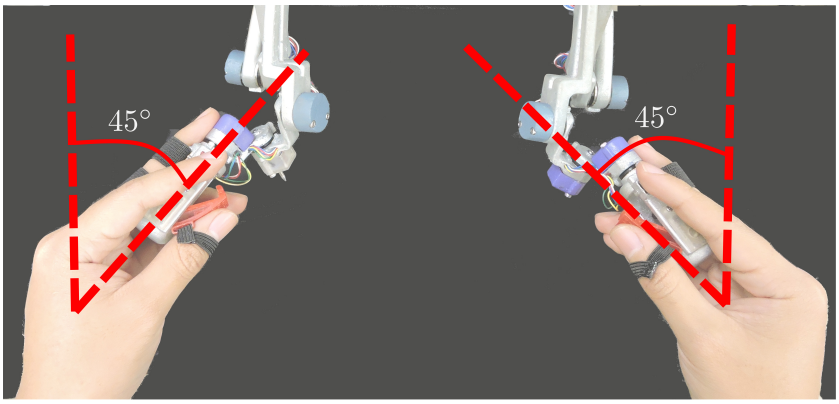

Figure 3.7: The neutral posture of the surgeon's hands having a 45° offset to provide comfortable operation.

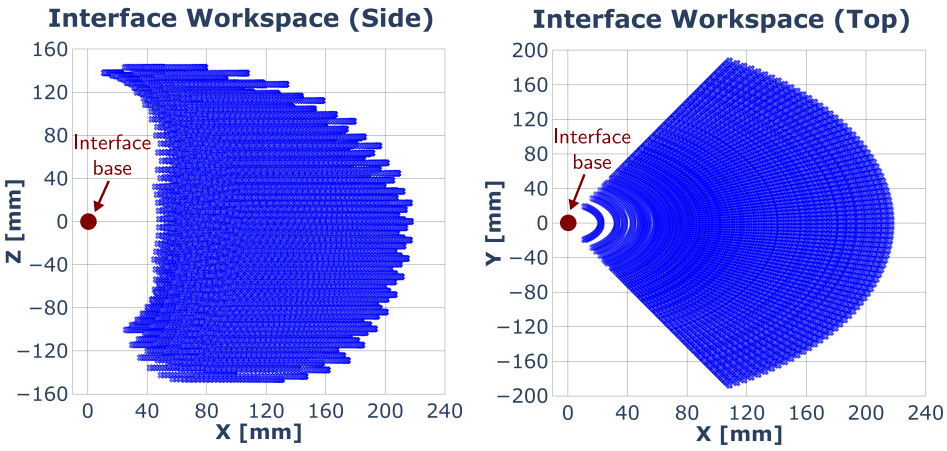

Figure 3.8: Serial-link mechanism workspace.

D-H Parameters

Link	a_i	α_i	d_i	θ_i
1	0	$\frac{\pi}{2}$	0	θ_1
2	l_1	0	d_1	θ_2
3	l_2	0	d_2	θ_3
4	l_3	0	d_3	θ_4
5	0	$\frac{\pi}{2}$	d_4	θ_5
6	0	$-\frac{\pi}{2}$	0	θ_6
7	0	0	0	θ_7

with:

$l_1 = 100$ mm
$l_2 = 82$ mm
$l_3 = 82$ mm

(Left)
$d_1 = 27$ mm
$d_2 = -11$ mm
$d_3 = -24.5$ mm
$d_4 = 29.5$ mm

(Right)
$d_1 = -27$ mm
$d_2 = 11$ mm
$d_3 = 24.5$ mm
$d_4 = -29.5$ mm

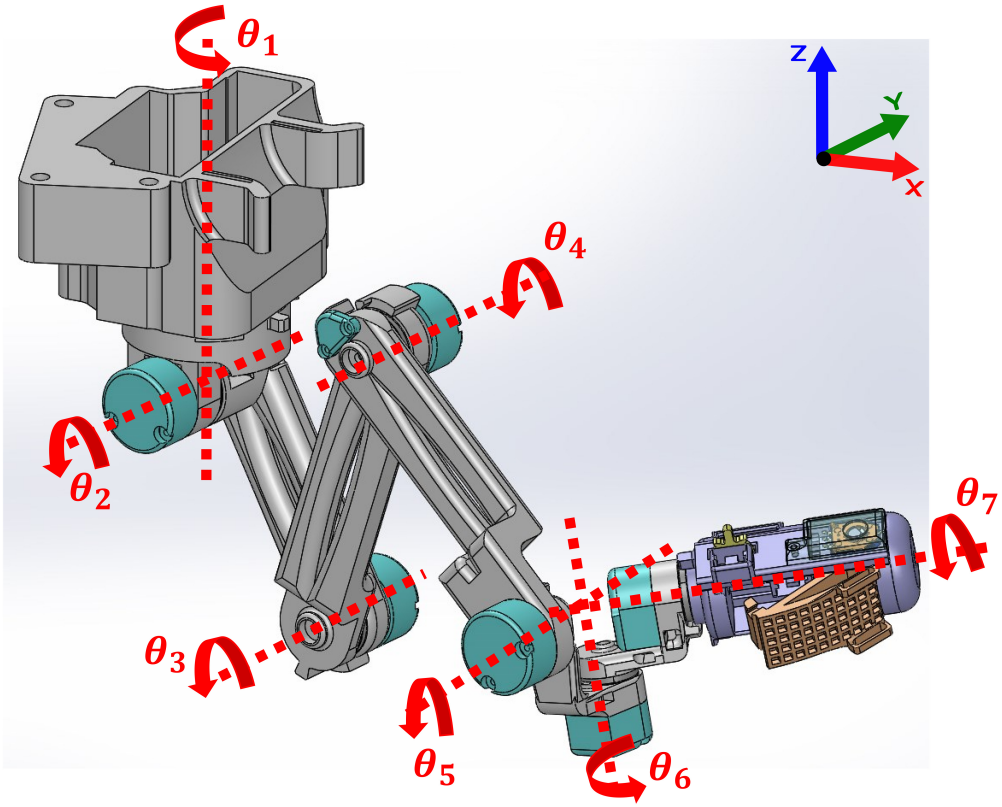

Figure 3.9: Kinematic structure of the serial-link mechanism.

Chapter 4

Robot motion control

The robot motion control varies depending on the EES subtask being performed, as mentioned in Section 2.3. Throughout this chapter, we provide a detailed description of the control algorithms implemented at each stage. Section 4.1 introduces the admittance control implemented to position the robot in a suitable initial position for tool insertion. A variable admittance control for tool insertion and extraction, and the virtual RCM implementation is discussed in Section 4.2. The control of the multi-DOF forceps simultaneously with the robot arm through the user interface is described in Section 4.3. Section 4.4. presents the low-level online trajectory generation. Finally, Section 4.5 discusses the experiments performed to evaluate the performance of the proposed methodology.

4.1 Positioning

To accomplish proper positioning of the articulated forceps before insertion, the robot followed the operator's desired motion through forces/torques exerted over the commanding handle attached to the robot. An admittance control scheme was used to enable compliant motion of the manipulator [100]. The implemented admittance controller measured the externally applied forces/torques via the 6-DOF F/T sensor and generated the target velocity from the following admittance model bounded within the saturation limits:

$$F_h = M_v \dot{V}_{ee} + D_v V_{ee} \tag{4.1}$$

where $F_h \in \Re^6$ is the external force/torque command, $V_{ee} \in \Re^6$ and $\dot{V}_{ee} \in \Re^6$ are the desired EE velocity and acceleration, respectively, and M_v and $D_v \in \Re^{6\times 6}$ are constant positive definite diagonal matrices that represent the desired virtual mass and damping, respectively. The velocity and force vectors consisted of linear and angular components,

respectively, as:

$$V_{ee} = \begin{bmatrix} \dot{p}_{ee} \\ \omega_{ee} \end{bmatrix}, \quad \text{and} \quad F_h = \begin{bmatrix} f_h \\ \tau_h \end{bmatrix} . \tag{4.2}$$

The desired pose of the robot was then obtained by integrating the forceps tip target velocity. Figure 4.1 depicts a block diagram of the proposed 6-DOF cooperative force controller based on admittance control. Given the force/torque command applied by the surgeon, f_h and τ_h, the discretized desired velocity of the forceps tip (EE) is obtained as follows:

$${}^{B}V_{ee[k]} = \frac{M_v {}^{B}V_{ee_{[k-1]}} + \Delta T {}^{B}F_{h[k]}}{M_v + \Delta T D_v} \tag{4.3}$$

where k is a discrete time step and ΔT is the control period. We can then obtain the translational motion of the forceps tip as:

$${}^{B}p_{ee[k]} = {}^{B}p_{ee[k-1]} + \Delta T {}^{B}\dot{p}_{ee[k]} . \tag{4.4}$$

For the rotational motion, we used the relationship between the time derivative of the rotation matrix R and the angular velocity ω,

$$\dot{R}_{ee} = S(\omega_{ee}) R_{ee} \tag{4.5}$$

where $S(\omega_{ee}) \in \Re^{3\times3}$ is the angular velocity tensor (skew-symmetric matrix). The desired orientation of the EE can be computed by integrating (4.5) as:

$${}^{B}R_{ee[k]} = {}^{B}R_{ee[k-1]} e^{S(\Delta T \omega_{ee[k]})} . \tag{4.6}$$

By using the Rodrigues rotation formula, we obtain:

$${}^{B}R_{ee[k]} = {}^{B}R_{ee[k-1]} (I + \sin\sigma \hat{S} + (1 - \cos\sigma)\hat{S}^2) \tag{4.7}$$

where $\hat{S}$ is the normalized skew-matrix and σ is the magnitude of the rotation vector $\theta = \Delta T \omega_{ee}$:

$$\sigma = \left\| \Delta T \omega_{ee} \right\|, \quad \text{and} \quad \hat{S} = \frac{S(\Delta T \omega_{ee})}{\sigma} . \tag{4.8}$$

Note that we ensured the orthogonality of the resulting rotation matrix with QR decomposition using Householder reflections.

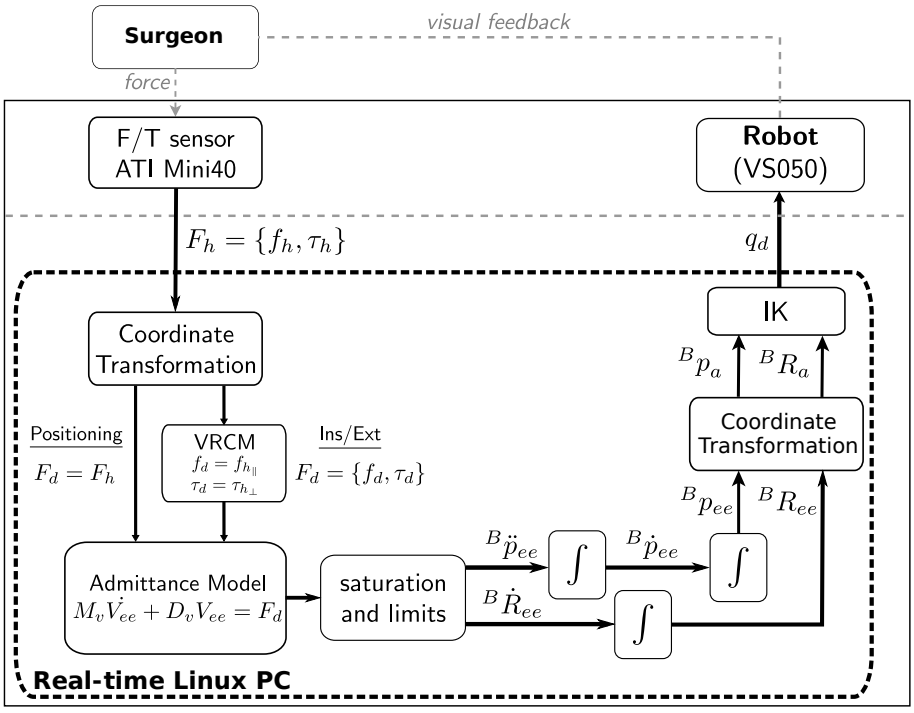

Figure 4.1: Block diagram of the force based cooperative control.

4.2 Insertion and extraction

During the insertion and extraction of the robotic forceps, the same admittance controller presented in Section 4.1 was used, but with two additional features to ensure the safe and stable motion of the forceps inside the patient's nasal cavity: variable admittance parameters and a virtual remote-center-of-motion (VRCM).

4.2.1 Variable admittance parameters

During the forceps insertion and extraction, a human-arm like motion was desired to provide intuitiveness and safe control of the forceps motion. Thus, a rapid movement was commonly performed close to the nostril area, while an accurate and slow positioning was preferred when reaching the target inside the sella region. In the admittance model, high precision and smooth motion could be achieved with high damping parameters, but large forces and time were needed. Low damping, on the other hand, required small forces and a short time, but the motion was less accurate. To this end, we implemented a variable admittance model, where the damping parameters in (4.1) were modified online in proportion to the distance between the forceps tip and the VRCM as:

$$D_v = D_{v_0} + \alpha \left\| p_{ee} - p_{vrcm} \right\| \tag{4.9}$$

with

$$\alpha = \begin{cases} \alpha_d & \text{if } ({}^{VRCM}p_{ee}^{\top}){}^{VRCM}u_x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.10}$$

where D_{v_0} is a constant initial damping matrix, α_d is a positive constant, and ${}^{VRCM}u_x$ is the unit vector along the forceps shaft .

4.2.2 Virtual remote-center-of-motion

To prevent damage to the surrounding tissues and enhance operation safety, we introduced a VRCM for specific transitions and task states, which was automatically activated at the transition (T_1) phase and enabled during the insertion (S_2), manipulation (S_3), and extraction (S_4) states. When the positioning (S_1) state ended, the forceps tip was at the entrance of the nostril, and this position was recorded and the VRCM position fixed around this point. To constrain the movement of the forceps through the VRCM, we used a hard virtual fixture [101]. The force command F_h was decomposed into the parallel ($F_{h_\parallel}$) and orthogonal ($F_{h_\perp}$) components to the forceps shaft axis (${}^{EE}u_x = {}^{VRCM}u_x$) as:

$${}^{B}F_h = {}^{B}F_{h_\parallel} + {}^{B}F_{h_\perp} \tag{4.11}$$

where:

$${}^{B}F_{h_\parallel} = {}^{B}F_h({}^{B}R_{VRCM}{}^{VRCM}u_x) \; . \tag{4.12}$$

We could now constrain the forceps motion with the use of a force/torque command (${}^{B}F_d$) defined by:

$${}^{B}F_d = \begin{bmatrix} f_d \\ \tau_d \end{bmatrix} = \begin{bmatrix} {}^{B}f_{h_\parallel} \\ {}^{B}\tau_h - {}^{B}\tau_{h_\parallel} \end{bmatrix} \; . \tag{4.13}$$

The parallel force component ${}^{B}f_{h_\parallel}$ and the orthogonal torque component ${}^{B}\tau_{h_\perp} = {}^{B}\tau_h - {}^{B}\tau_{h_\parallel}$ were selected to generate a translational velocity command along the direction of ${}^{VRCM}u_x$ and to remove rotational motions around the same axis ${}^{VRCM}u_x$. As a result, the forceps motion was constrained within 3-DOF, only allowed to generate inclinations pivoted over the VRCM and translations along the shaft axis.

4.3 Manipulation

In the manipulation state, it was assumed that the forceps tip was situated in the target area. The multi-DOF forceps and the robot arm motion were controlled simultaneously by the 7-DOF serial-link interface. To match the workspace, the reference frame of the interface ${}^{B}p_{h_{ref}} = {}^{B}R_{I}{}^{I}p_{h_{ref}} + {}^{B}p_{r_{ref}}$ and the target position ${}^{B}p_{t}$ were registered during the transition T_3 by pressing the push button located at the top of the interface gripper and using the EE position as a reference ${}^{B}p_{g} = {}^{B}p_{ee}$.

Figure 4.2 depicts the block diagram of the master-slave controller achieving simultaneous control of the robot arm and the forceps tip by the serial-link interface for the manipulation state. The surgeon's hand pose ${}^{B}X_{h} = \{{}^{B}p_{h}, {}^{B}R_{h}\}$ was obtained from the interface, and the robot kinematics was computed at each control cycle. The master-slave controller scaled the surgeon's hand displacement ΔX_{h} by a factor of K_{s} and generated the desired displacement of the forceps tip pose ΔX_{ee} with respect to the target reference. We then obtained the desired EE pose, described as:

$$ {}^{B}p_{ee} = {}^{B}p_{g} + K_{s}[{}^{B}R_{I}({}^{I}p_{h} - {}^{I}p_{h_{ref}}) + ({}^{B}p_{r} - {}^{B}p_{r_{ref}})] \tag{4.14} $$

$$ {}^{B}R_{ee} = {}^{B}R_{I}({}^{I}R_{h}{}^{I}R_{h_{ref}})^{\top}{}^{B}R_{I}{}^{\top}{}^{B}R_{g} \, . \tag{4.15} $$

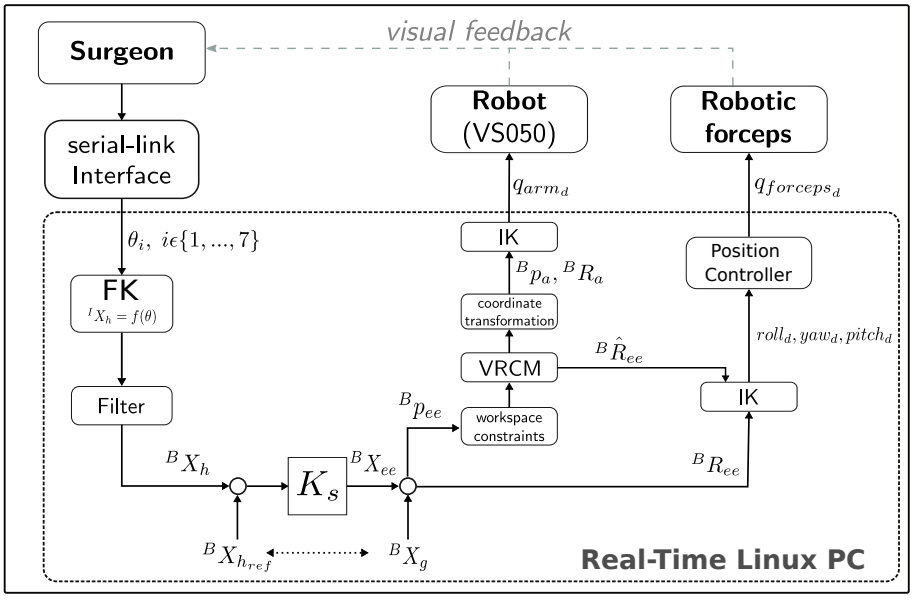

Figure 4.2: Block diagram of the master-slave controller.

To ensure the safety of the robot's motion, two active constraints were implemented: workspace boundaries (virtual walls) and a VRCM.

Workspace virtual walls

With the workspace virtual walls, the forceps tip workspace was constrained within a cuboid with dimensions 40 × 25 × 40 mm, as shown in Figure 4.3a. Compared to other common workspace constraints (e.g., sphere or prism), a cuboid is simple to implement for faster computation and could provide a wider workspace for a knot tying task, in which the thread must be tightened by pulling the end of the suture using the forceps. The desired position of the forceps tip in the target frame ${}^{G}p_{ee} = {}^{G}R_{B}{}^{B}p_{ee} + {}^{G}p_{b}$ was projected inside the cuboid as:

$$ {}^{G}p_{ee_i}\underset{i\in\{x,y,z\}}{} = \begin{cases} \ell_i^{min} & {}^{G}p_i \leq \ell_i^{min} \\ \ell_i^{max} & {}^{G}p_i \geq \ell_i^{max} \\ {}^{G}p_{ee_i} & \text{otherwise} \end{cases} \tag{4.16} $$

where ℓ_i^{min} and ℓ_i^{max} are the workspace predefined minimum and maximum limits, respectively. We then applied the inverse operation ${}^{B}p_{ee} = {}^{G}R_{B}{}^{\top}({}^{G}p_{ee} - {}^{G}p_{b})$ to obtain again the desired forceps tip position in the base frame.

Virtual Remote Center of Motion (VRCM)

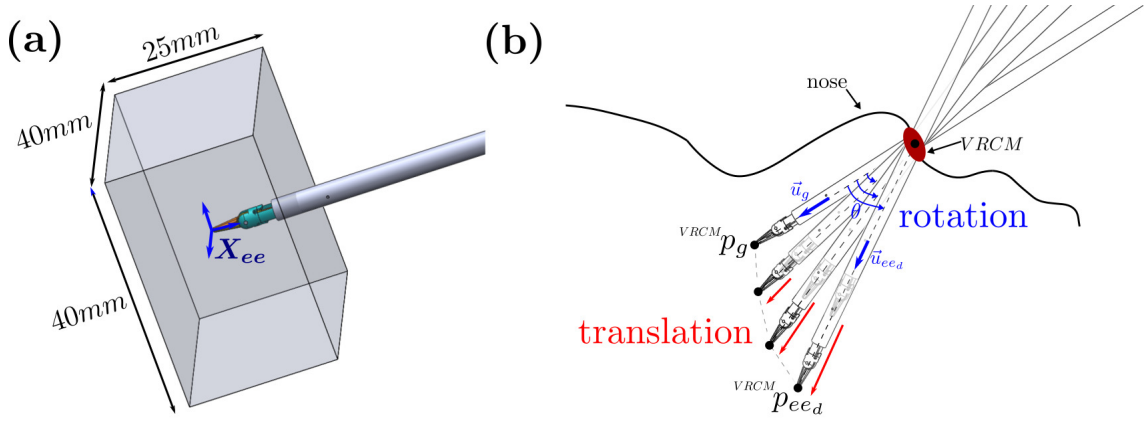

Figure 4.3: Virtual constraints. (**a**) Workspace virtual walls specified by a cuboid. (**b**) VRCM constraint implemented as a simultaneous rotation and translation along the VRCM frame.

Once the desired pose of the forceps tip X_{ee} was obtained, it was then split into a desired position command p_{ee_d} and an orientation command R_{ee_d} for a two-level controller, giving priority to preserve the VRCM constraint defined in Section 4.2. We followed a simultaneous rotation-translation sequence [102] to reach the desired EE position p_{ee}, as shown in Figure 4.3b.

First, we computed the unit vectors ${}^{VRCM}u_g = \frac{{}^{VRCM}p_g}{\|{}^{VRCM}p_g\|}$, ${}^{VRCM}u_{ee_d} = \frac{{}^{VRCM}p_{ee_d}}{\|{}^{VRCM}p_{ee_d}\|}$ from the target position and desired EE position, respectively. Next, the rotation operation to obtain ${}^{VRCM}u_{ee_d}$ from ${}^{VRCM}u_g$ was represented as an angle-axis operation composed of an axis of rotation v and the angle of rotation θ, defined as:

$$ {}^{VRCM}v = {}^{VRCM}u_g \times {}^{VRCM}u_{ee_d}, \quad \text{and} \quad \theta = \arccos(({}^{VRCM}u_g{}^{\top}){}^{VRCM}u_{ee_d}) \; . \tag{4.17} $$

The same rotation was applied over the target frame to obtain the partial forceps tip orientation ${}^{B}\hat{R}_{ee}$:

$$ {}^{B}\hat{R}_{ee} = {}^{B}R_{(v,\theta)}{}^{B}R_{ee} \; . \tag{4.18} $$

To reach the desired orientation command ${}^{B}R_{ee}$, the additional DOF from the active forces were used. The complementary orientation is obtained as:

$$ {}^{B}R_{forceps} = {}^{B}R_{ee}({}^{B}\hat{R}_{ee}{}^{\top}) \Rightarrow (roll_d, yaw_d, pitch_d) \; . \tag{4.19} $$

From the rotation matrix, an Euler angle representation $(roll_d, yaw_d, pitch_d)$ was generated for the multi-DOF forceps within the joints' range of motion and projected onto the available DOF of the forceps (2-DOF: roll and grasp, 4-DOF: roll, yaw, pitch, and grasping).

The implementation of the VRCM was experimentally verified by a tracing task in which the forceps tip followed a trajectory defined by a circumference with a diameter of 2 cm. A magnetic motion capture (Aurora, Northern Digital Inc., Ontario, Canada) was used to track the position of a magnetic sensor placed over the forceps shaft where the VRCM was defined. We measured the RMS error between the position of the sensor and the VRCM, and we obtained a maximum RMS error of 1.59 mm and a median of 0.35 mm.

4.4 Online trajectory generation

The robot arm desired pose ${}^{B}X_a = \{{}^{B}p_a, {}^{B}R_a\}$ is computed by frame transformations as:

$$ {}^{B}p_a = {}^{B}p_{ee} - {}^{B}R_A{}^{A}p_{ee} \tag{4.20} $$

and:

$$ {}^{B}R_{a} = {}^{B}R_{ee}\,{}^{B}R_{EE}{}^{\top} . \tag{4.21} $$

A smooth Cartesian EE trajectory was generated online by using the Reflexxes Type II Motion Library [103]. This online trajectory generator (OLT) provides commands for low-level motion control in real-time with position and velocity constraints. The TRAC-IK library [104] was used to enforce the minimal configuration change combining two inverse kinematics approaches, pseudo-inverse Jacobian (4.22) and SQP-SS (4.23), which can be computed within the control loop of 500 Hz as:

$$ q_{[k]} = q_{[k-1]} + J^{\dagger} p_{err} \tag{4.22} $$

$$ \begin{aligned} &\arg\min_{q \in R^{6}} p_{err}^{\top}\ p_{err} \\ s.t. \quad & q \leq b_i, \quad i = 1, ..., 6 \end{aligned} \tag{4.23} $$

where q represents the joint position, p_{err} the Cartesian error vector, and b_i the joint limits. For the transition to a halt state (S_5), we used the current robot pose and null velocity as a target for the OLT. When the velocity reached zero, the halt state was enabled. For transitions between other states, the OLT ensured smooth transition with continuous position and velocity.

4.5 Experiments and discussion

To evaluate the performance of our proposed scheme, we designed four testing experiments oriented toward testing reachability, manipulability, and dexterity during common surgical tasks. We compared it with the use of conventional forceps used in endoscopic endonasal surgery. The following four tasks were performed by the participants using both conventional surgical instruments and the proposed cooperative interface: (1) a reachability task to test the ability of the robot to reach anatomic areas of interest in endonasal surgery such as pituitary tumors in the sellar area, (2) a pick-and-place task to determine if the users can manipulate objects and translate them dexterously in a constrained environment, (3) a block-in-the-hole task to add the need of orientation adjustments in object manipulation, and (4) a needle stitching task to evaluate the performance in a common surgical task in endonasal surgery. All subjects gave their informed consent for inclusion before they participated in the study. The study was approved by the Ethical Research Committee of Nagoya University.

4.5.1 Experiment 1: Reachability

The aim of this experiment was to determine if potential users could operate the robotic system to position and insert the articulated forceps satisfactorily within the anatomical constraints of the nasal cavity to reach an area of interest. The experimental setup is presented in Figure 4.4.

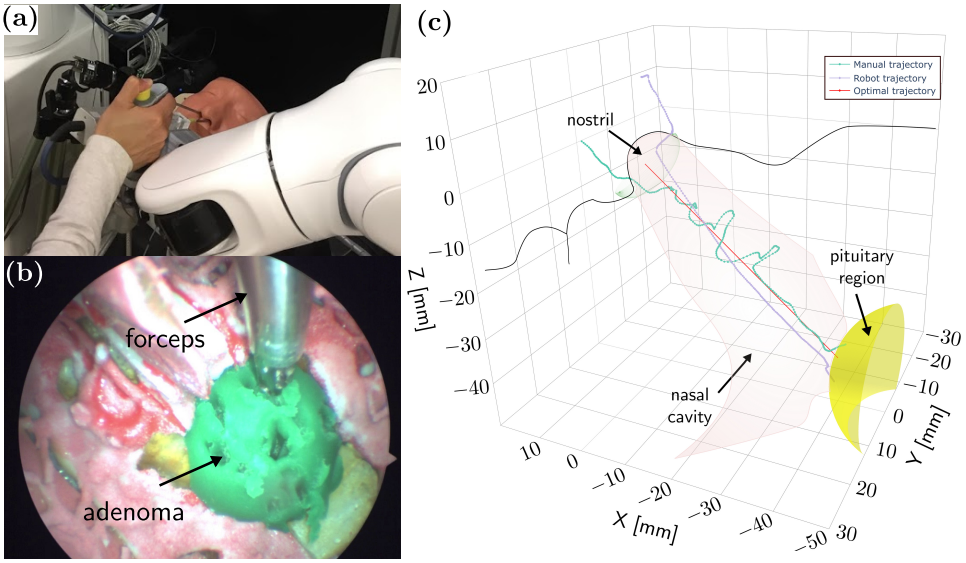

Figure 4.4: (**a**) Reachability experiment setup. (**b**) Adenoma model in the pituitary region represented with a green mark in the human head phantom. (**c**) Forceps tip trajectory comparison between manual and robot operation.

We used a human head phantom (M01-SET-TSPS-E1, SurgTrainer Ltd., Ibaraki, Japan), prepared by a surgeon to expose the sphenoid sinus, and the area of interest was a pituitary adenoma model represented with a green mark (see Figure 4.4b). The head phantom was rigidly attached to a support and slightly tilted to resemble a normal clinical setup. A rigid endoscope (30°, 2.7 mm diameter) was used for visualization inside the head phantom and targeting the area of interest. The participants were asked to introduce a surgical tool into the left nostril and reach the area of interest with the use of conventional surgical instruments and with the proposed robotic system. A magnetic motion capture system (Aurora, Northern Digital Inc., Ontario, Canada) was utilized to measure the motions of the surgical instruments for both the manual and robot task operation. Five subjects between the ages of 20 and 35 participated in this experiment. We defined the optimal trajectory as the line connecting the center of the nostril and the center of the target pituitary region. The root mean square error (RMSE) of the distance between the trajectory performed and its

projection over the optimal trajectory unit vector $u_{opt} = \frac{x_{target} - x_{nostril}}{\|x_{target} - x_{nostril}\|}$ was used as metric to evaluate the system performance as:

$$RMSE = \sqrt{\sum_{i=1}^{n_{samples}} \frac{\|(x - (x^\top u_{opt})u_{opt})\|^2}{n_{samples}}} \,. \tag{4.24}$$

A representative trajectory is shown in Figure 4.4c. The RMSE is shown in Figure 4.5a. The trajectory obtained with manual operation showed frequent direction changes and oscillations because of hand tremor. With the use of the proposed robotic system, the trajectory became smoother, and the deviation error was reduced by 30%. The time to complete the task is shown in Figure 4.5c. Although a velocity limitation of 10 mm/s was imposed on the robot, the completion time showed a 20% reduction with the proposed system compared to a manual insertion. The virtual remote-center-of-motion constrained the forceps motion to avoid collision with the surrounded tissue, and the variable damping in the force control provided stable and smooth motion control while approaching the target. In a real EES operation, tool exchange is a frequent task. The results presented indicate a potentially significant reduction of the time spent on this task when using the proposed robotic system. Additionally, although the RMSE remained close for both types of operation, it is important to observe from Figure 4.5b, that the maximum deviation from the optimal trajectory obtained for the robotic operation was about 7 mm, while for the manual operation it was about 12 mm. In a real surgical scenario, with anatomic constraints imposed along the nasal cavity, the robotic system will greatly reduce potential harm to the surrounded tissues.

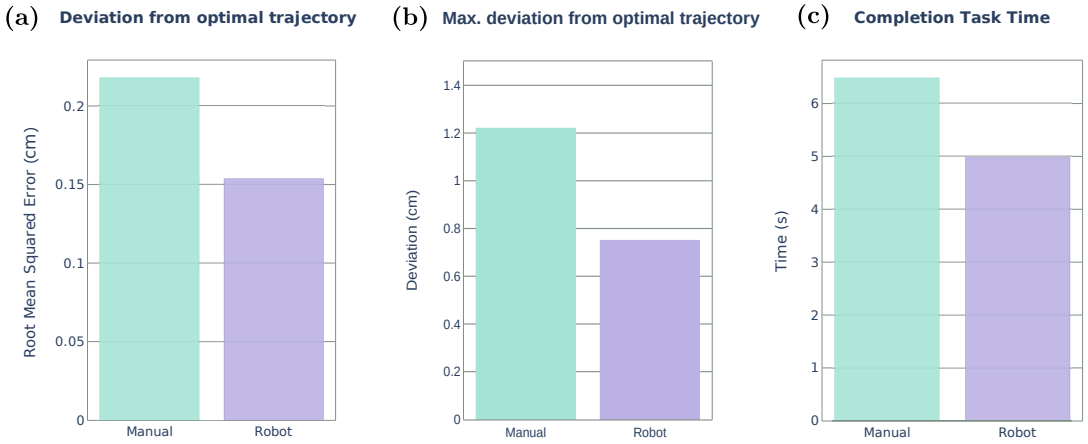

Figure 4.5: (**a**) RMSE between the trajectory performed and the optimal trajectory. (**b**) Maximum deviation from the optimal trajectory. (**c**) Task completion time.

4.5.2 Experiment 2: Pick-and-place and Experiment 3: Block-in-hole

In the pick-and-place task, the forceps were employed to grasp four silicon tubes (inside diameter: 3 mm, outside diameter: 5 mm, length: 8 mm) and place them on numbered pegs (see Figure 4.6b). The testbed was placed 100 mm away from the nostril. Right and left hands were alternated for each silicon tube. The task was repeated three times for each hand with both the conventional instruments and the proposed cooperative user interface. The block-in-hole task was similar to the pick-and-place task with an additional need for orientation adjustments while positioning the blocks. The forceps were employed to grasp three blocks (two 5 mm edge cubes and one 4 mm edge tetrahedron) and place them on the corresponding hole located under the initial hole, with a similar shape, but different orientation (see Figure 4.6c).

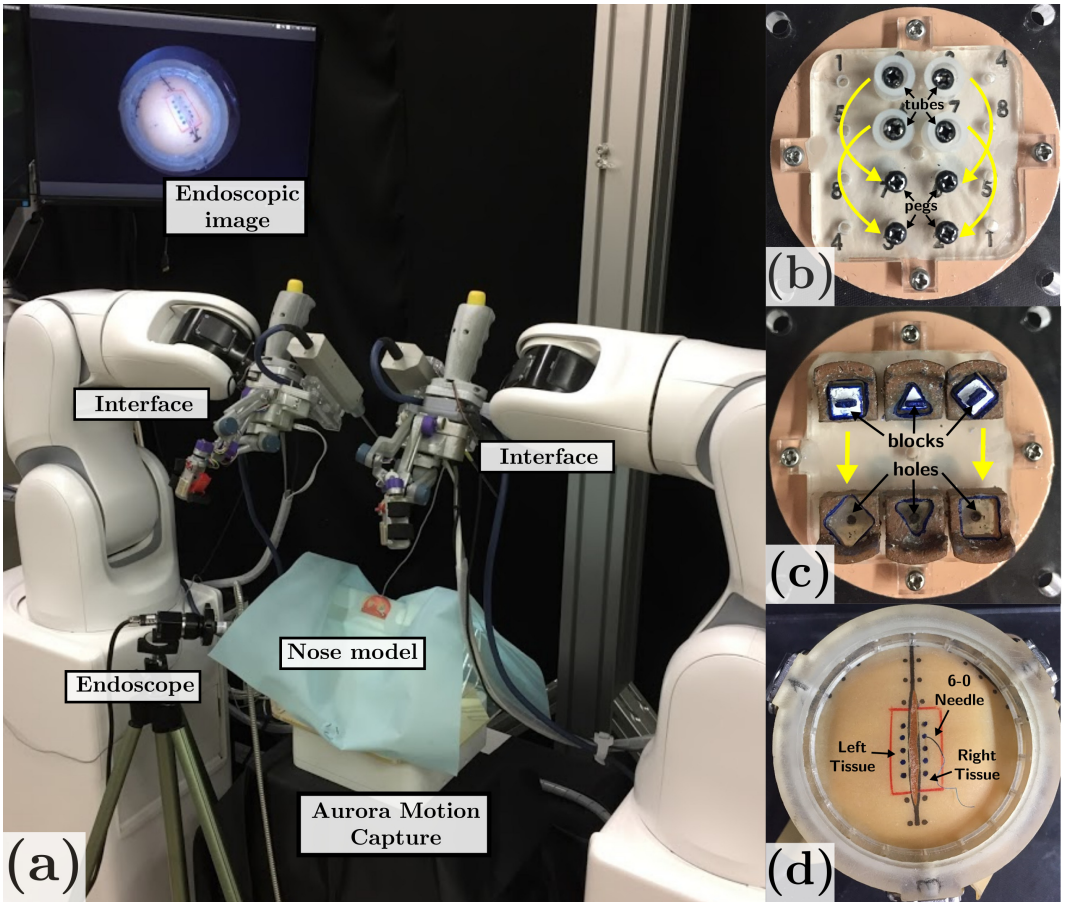

Figure 4.6: (**a**) Experiment setup. (**b**) Pick-and-place testbed. (**c**) Block-in-hole testbed. (**d**) Stitching testbed.

The testbed was located 100 mm away from the nostril. Right and left hands were alternated for each block. The task was repeated three times for each participant with both the conventional instruments and the proposed cooperative user interface.

For the pick-and-place and block-in-hole tasks, a nose phantom was used (shown in Figure 4.6a). It comprised a 3D printed nose model and an acrylic platform for the testbeds. A magnetic motion capture system (Aurora, Northern Digital Inc., Ontario, Canada) was used to measure the motions of the surgical instrument for both the manual and robot task operation for evaluation purposes. For each trial, we considered the following aspects to evaluate the performance:

- Task completion time (s): starting from the first contact with the tube/block until the release of the last tube/block.

- Motion smoothness: we used the root mean squared jerk (RMSJ) [105] as a metric, defined by:

$$RMSJ = \sqrt{\frac{1}{t_1 - t_0} \int_{t=t_0}^{t=t_1} \|\dddot{x}\|^2 dt} \,. \tag{4.25}$$

Eight subjects between the ages of 20 and 35 who had no previous surgical training took part in these experiments. Before the experiments, each participant was instructed to practice for 5–10 minutes until they were familiarized with the operation. The results for the pick-and-place task are depicted in Figure 4.7.

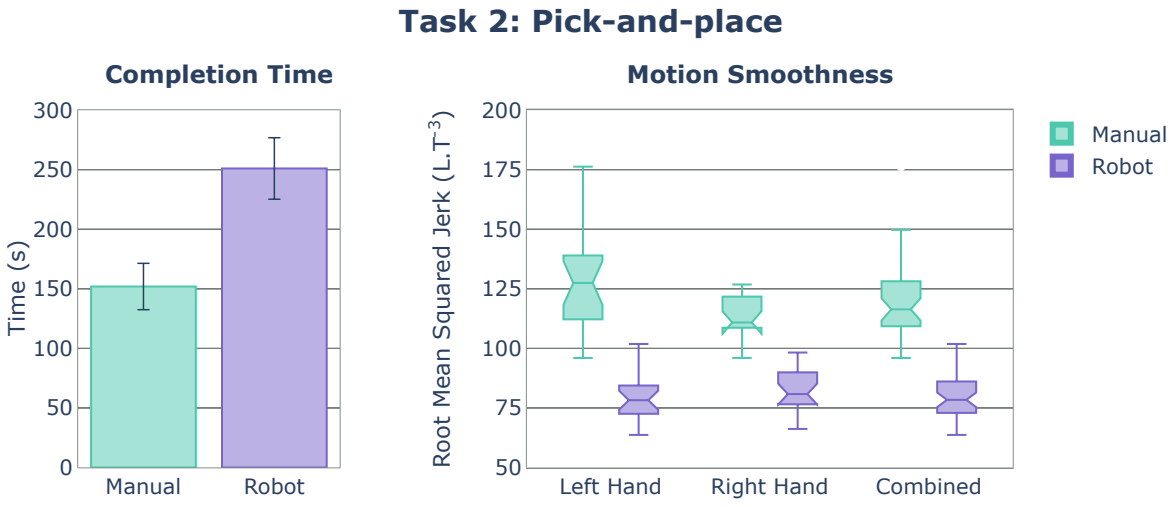

Figure 4.7: Experimental results of the pick-and-place task.

The time to complete the pick-and-place task was approximately 66% longer with the use of our proposed system than the case of the manual operation of conventional forceps. This was expected due to the velocity limitation of 10 mm/s imposed on the forceps tip motion on the robot arm. Despite this, the root mean square jerk, used as a metric of smoothness, showed approximately a 20% reduction of the variability and mean values when using our proposed system. A higher dexterity in the right hand compared with the left hand was observed as lower variability in the manual operation. In the case of the robotic system, both cases showed a similar variability, which indicated that similar levels of dexterity could be rapidly achieved for both hands. Furthermore, the notches in the combined box represents the 95% confidence interval for the median. In our results, these confidence intervals did not show any overlap, which indicates that the true value of the motion smoothness metric median for the robotic system is higher to the use of conventional instruments with a confidence of 95%.

The results for the block-in-hole are depicted in Figure 4.8. Similar to the pick-and-place task, the time to complete the block-in-hole task was approximately 66% longer with the use of our proposed system due to the velocity limitation. However, in this case, the root mean squared jerk showed more than 40% reductions of the mean values when using our proposed system on both hands. Again, the combined box confidence intervals did not show any overlap, which indicated that the robotic system was able to provide a smoother motion (with a confidence of 95%) compared to the use of conventional instruments when orientation adjustments and higher precision were required.

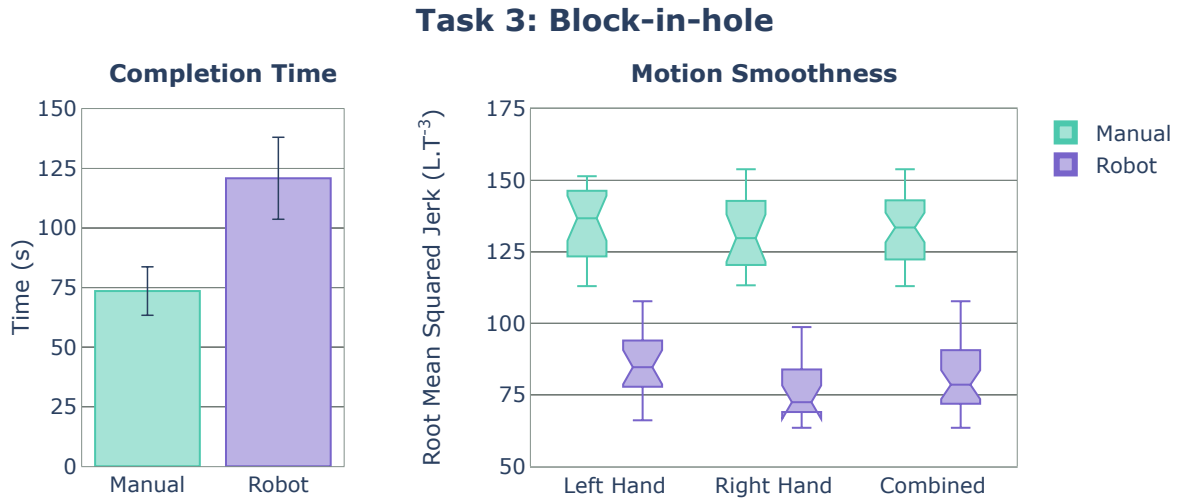

Figure 4.8: Experimental results of the block-in-hole task.

4.5.3 Experiment 4: Needle stitching task

The stitching task was intended to replicate a common task in endonasal surgery, where suturing of the dura matter is performed to prevent the leaking of cerebrospinal fluid. The experimental setup is shown in Figure 4.6a with the testbed shown in Figure 4.6d. The same nose model of Experiment 2 and Experiment 3 was used for this task. A force sensor (Nano17, ATI Industrial Automation, North Carolina, USA) was placed behind the elastic tissue to register the force applied during the needle manipulation.

The task consisted of grasping a 6-0 surgical needle and puncturing an elastic tissue with a 6-0 surgical needle. The participants were asked to follow a five-step procedure: (1) remove the needle from its initial position, and insert it into a mark on the right tissue using the right forceps (R-Insertion); (2) extract the needle from under the tissue (R-Extraction); (3) hold the needle using the right forceps with a proper orientation (Regrasping); (4) insert the needle from under the left tissue (L-Insertion); (5) extract the needle using the left forceps (L-Extraction). A total of three series of the full task were requested of each participant. The needle was initially inserted in the tissue and had to be grasped with the right forceps for the insertion and the left forceps for the final extraction. Five subjects between the ages of 20 and 35 who had no previous surgical training participated in these experiments. Before the experiments, each participant was instructed about the procedure and practiced for up to 15 minutes until they were familiarized with the operation. We used both the 2-DOF forceps and 4-DOF forceps for this experiment.

For each trial, we considered the following aspects to evaluate the task performance:

- Task completion time (s): starting from removing the needle from its initial position until the complete needle extraction on the left side.
- Interaction force (N): recorded with a force sensor placed behind the tissue.

The resulting task completion times are depicted in Figure 4.9. Using the proposed robotic system with the 2-DOF forceps exhibited a longer time, but with a shorter confidence interval for all steps, except for the Regrasping step. Passing the needle with a proper orientation from one forceps to another required precise control of the needle and was challenging to achieve without training. When using our proposed system, the time to perform this task was reduced by approximately 50%. With the robotic system and the 4-DOF forceps, the completion time was reduced for the R-Insertion and L-Insertion steps, and the Regrasping step was no longer needed because the R-Extraction could be performed easily with the right forceps thanks to the two additional DOFs. For the R-Insertion and L-Extraction steps, the operation with both articulated forceps showed a

higher completion time, which may be caused by the limitation in the grasping force of the articulated forceps used.

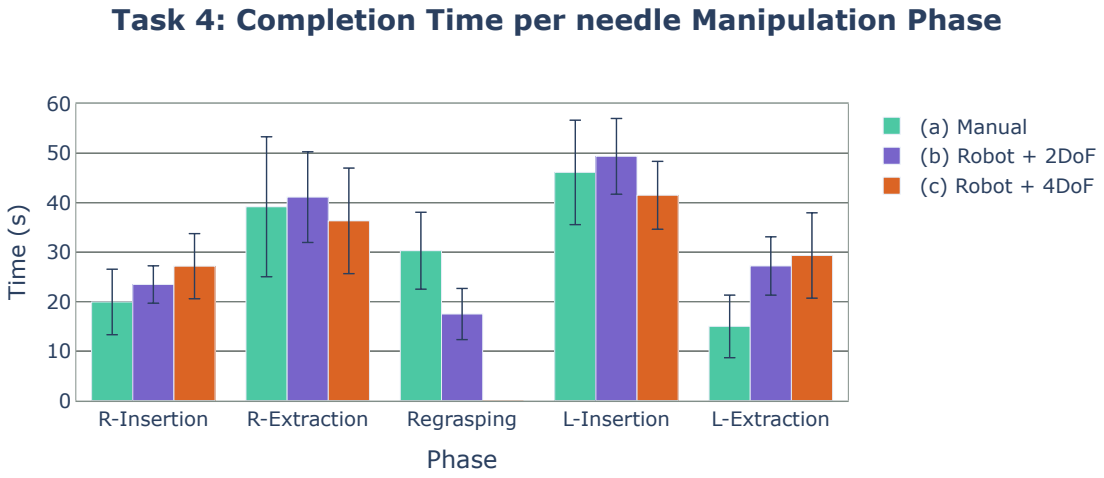

Figure 4.9: Completion time for needle stitching per phases with the use of (a) regular forceps, (b) the robot with 2-DOF forceps, and (c) the robot with 4-DOF forceps. Note that in the case of (c) the robot + 4 DOF, the Regrasping phase is not needed with the benefit of the dexterity of the articulated forces, reducing the task completion time significantly.

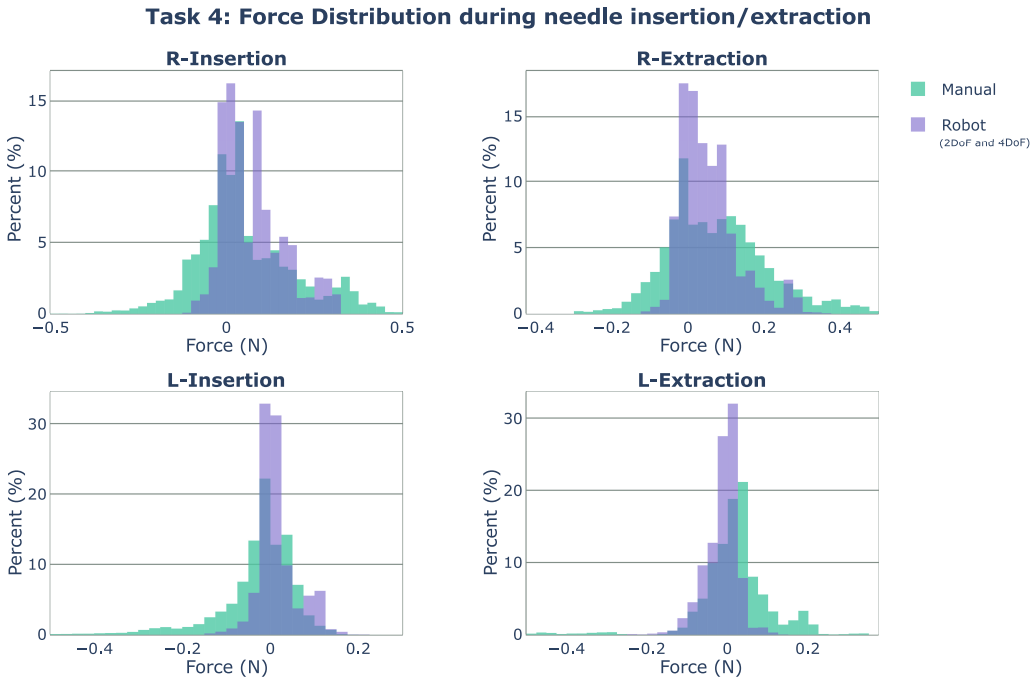

Figure 4.10: Force distribution during the needle insertion/extraction.

Figure 4.10 shows the force distribution resulting from the forceps interaction with the tissue during the needle insertion/extraction. The larger the number of bins close to 0 N, the less force applied to the tissue (less potential damage). The use of the robotic system (considering the use of both 2-DOF and 4-DOF forceps) reduced the range of applied force to almost half compared with the use of conventional forceps, from around [-0.4 N,0.4 N] to [-0.2 N, 0.2 N].

Finally, a complete stitching sequence achieved with the robotic system is shown in Figure 4.11.

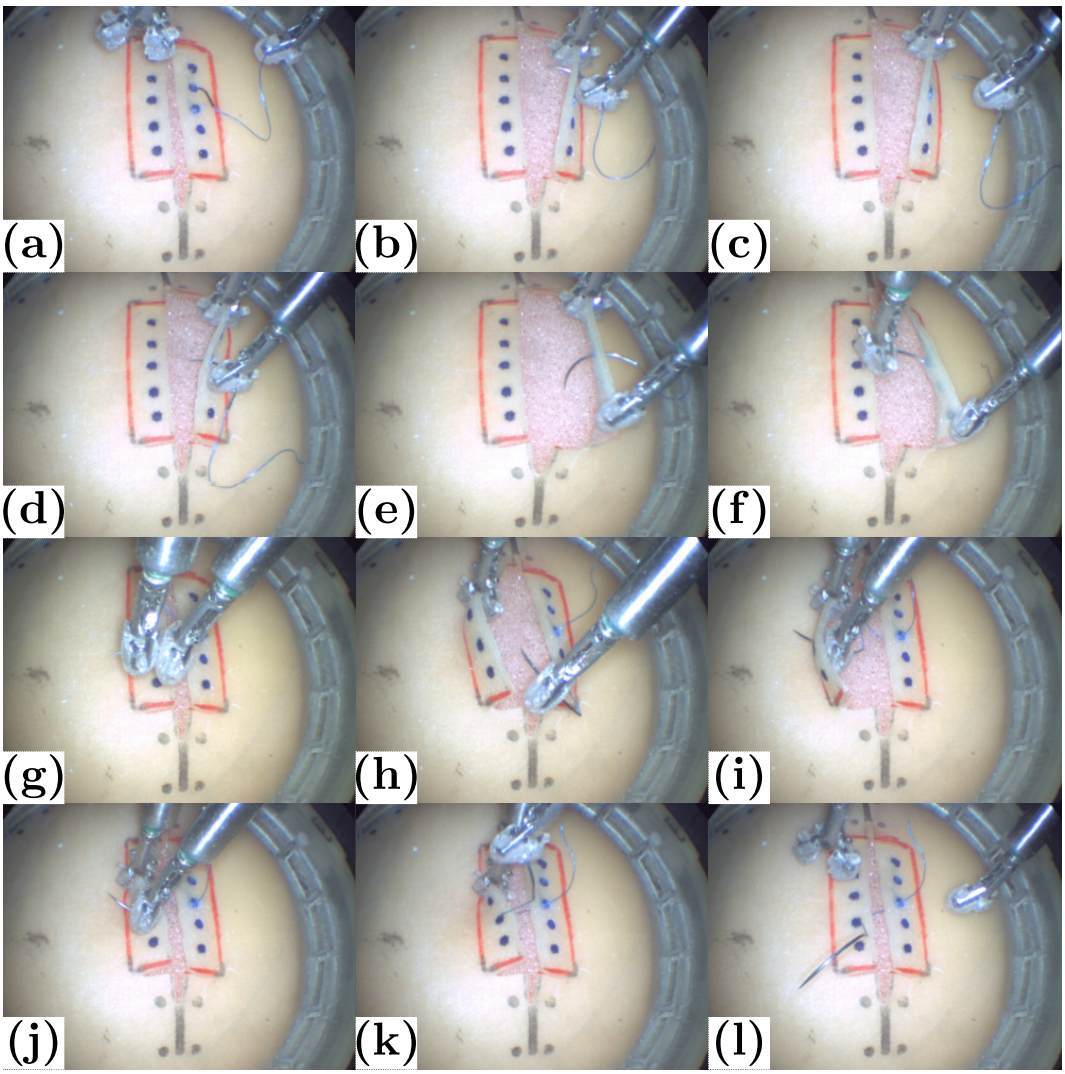

Figure 4.11: Needle stitching sequence.

Chapter 5

Robotic-assisted stitching

In the previous chapters, we have described a human-robot interface for controlling multi-DOF articulated forceps in EES (Chapter 3), and a multi-level motion control for specific task assistance (Chapter 4). However, the proposed approach can be further extended with the implementation of semi-autonomous execution of complex and repetitive surgical tasks. This chapter presents a robot-assisted stitching framework based on optimal needle trajectory generation subject to the stitching kinematic constraints, and a dual concurrent constrained control algorithm to track the generated trajectory. Section 5.1 describes the characterization of the stitching task. The optimization framework is presented in Section 5.2. Section 5.3 describes the constrained motion planner used to restrain and guide the motion of the robot during the stitching task. Section 5.4 discusses the performance evaluation of the proposed method.

5.1 Stitching Task characterization

5.1.1 Suturing in EES

Suturing is a challenging task in EES. It is usually performed to secure the dura after adenoma resection as shown in Figure 5.1. Typically, a 6-0 3/8 11mm circular needle is used, and two forceps are used simultaneously to accomplish a stitching task. The hand tremor hinders accurate needle positioning, and the tissue-needle interaction produces deviations in needle orientation, which makes stitching quite challenging and high-stressful for surgeons.

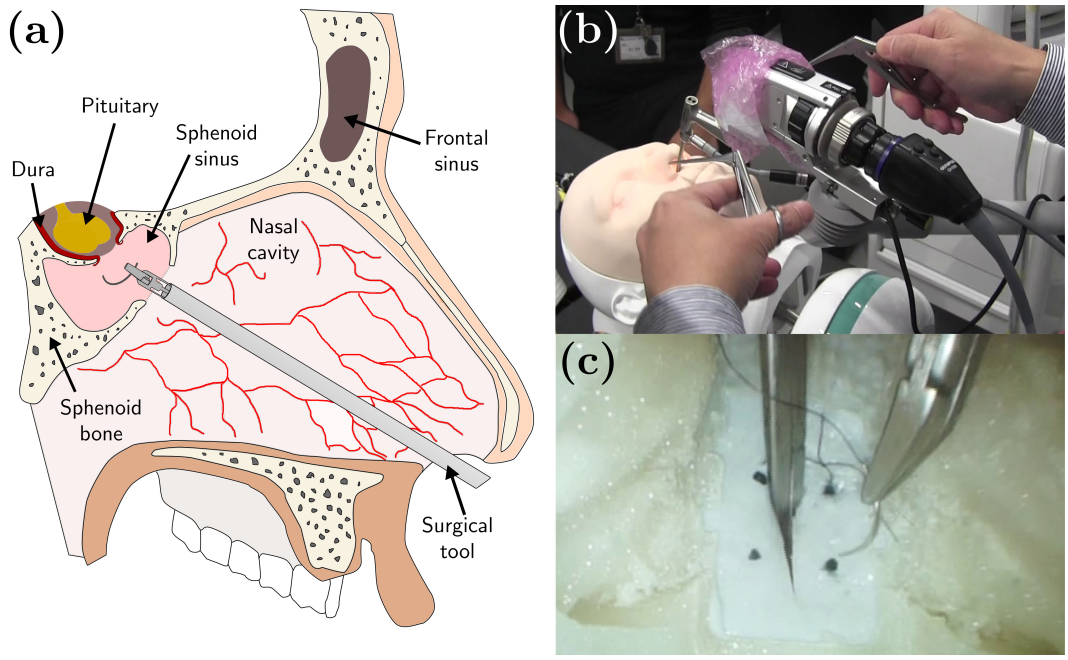

Figure 5.1: (**a**) Endoscopic endonasal surgery. (**b**) Dura suturing in EES performed in a head model phantom. (**c**) An example of an endoscopic view of the suturing task in the head model.

5.1.2 Description of the stitching workspace frames

The coordinate frames for the robotic system are defined as shown in Figure 5.2.

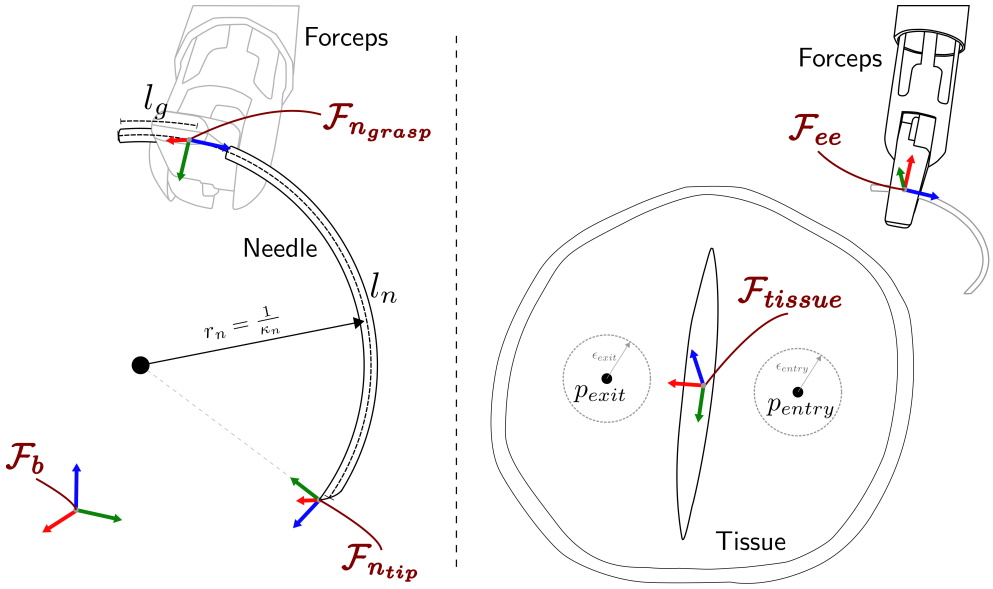

Figure 5.2: Definition of the coordinate frames used to represent the stitching task

F_b represents the robot arm base frame, F_{ee} is the forceps tip frame, $F_{n_{tip}}$ is the needle tip frame, $F_{n_{grasp}}$ is the frame associated to the needle grasping point, and F_{tissue} corresponds to the tissue frame with the z-axis aligned to the tissue surface normal vector.

5.1.3 Algorithm overview

The high-level schematic overview of the proposed algorithm is shown in Figure 5.3.

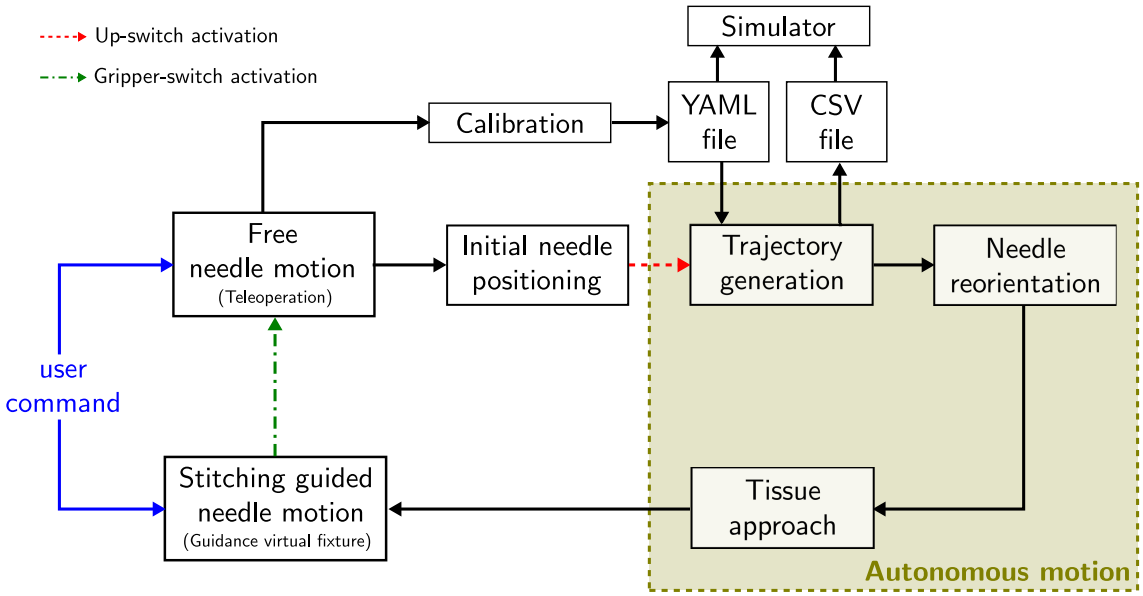

Figure 5.3: High level schematic overview of the proposed robot-assisted stitching.

The proposed algorithm assumes that the forceps have been inserted into the nasal cavity through a remote center of motion (RCM) placed over the patient nostril [106], which is kept fixed by the constrained motion planner during the entire execution of the stitching task. First, the surgeon can command the forceps tip pose through the user interface in a teleoperation mode. An initial calibration procedure can be performed to define the desired entry and exit points and to determine the orientation plane of the tissue surface, by positioning the forceps tip over the desired points and recording them in a YAML file. The calibration file can be either generated online or retrieved from the previously recorded ones at any time during the trajectory generation process. During the teleoperation mode, the needle pose can be freely controlled. The surgeon places the needle in a suitable initial position and activates the robot-assisted stitching sequence by pressing the gripper-button located in the user interface handle (see Figure 5.4).

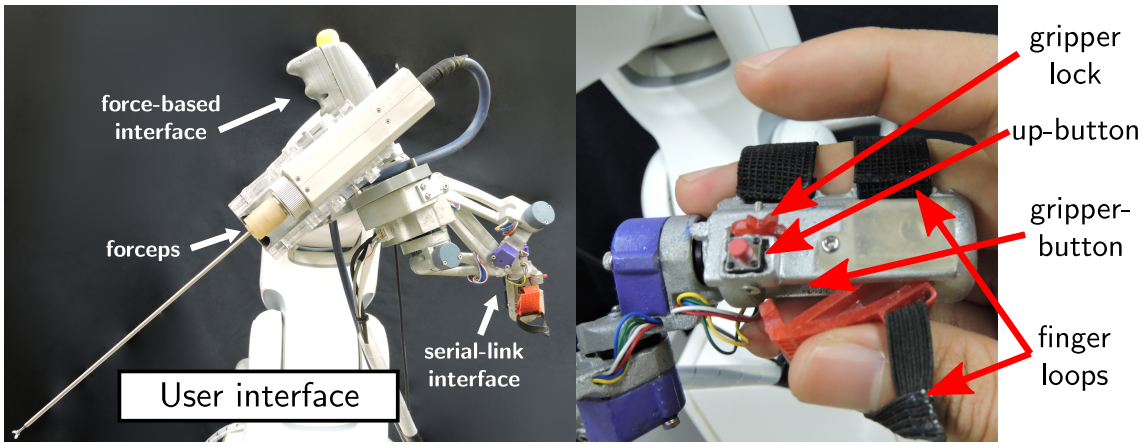

Figure 5.4: Push-buttons used to alternate between robot-assisted modes. The gripper handle contains two push-buttons: one above the handle (up-button), and the second one placed inside the gripper (gripper-button) The upper push-button enables the free motion control of the forceps tip and the gripper push-button activates the robot-assisted stitching mode.

The robotic system will automatically generate an optimal trajectory to be followed by the needle tip during the stitching task. The generated trajectory is stored as a CSV file that is synchronized with a simulator for trajectory visualization. The optimal trajectory defines the desired initial needle tip pose close to the entry point over the tissue. The robot reorients the needle and approaches to the desired initial needle pose following the trajectory defined by a linear interpolation for the translational motion and a spherical linear interpolation for the rotational motion. When the needle is expected to be in initial contact with the tissue, the robot stops and the stitching guided mode is activated. Now the robotic system follows a shared control scheme, in which the robot ensures that the needle tip pose (position and orientation) follows the optimal trajectory generated by the proposed algorithm, whilst the surgeon is in charge of the control of the insertion speed and direction of the motion. In this way, the optimal trajectory works as a virtual guidance constraint for the surgeon's command. In case of changes in the trajectory generation parameters, while performing the stitching sequence (e.g., changes in the desired entry or exit positions), the needle can be retracted along the optimal trajectory until full extraction, and a new trajectory can be generated with the updated parameters. Moreover, the surgeon can return to the free needle motion at any time by pressing the up-button located in the user interface handle. Then, the guidance virtual constraint is disabled and the surgeon can freely manipulate the needle pose.

5.2 Online optimization-based trajectory generation

The notation used in this section is listed in Table 5.1.

Table 5.1: The list of notation used in this thesis

Notation	Description
$\mu^{(k)}$	k-iteration penalty
$\delta^{(k)}$	k-iteration trust region size
$u^{(k)}$	k-iteration control variable
η_r	step rejection threshold
η_a	step acceptance threshold
γ_s	trust region shrinkage factor
γ_e	trust region expansion factor
μ_s	penalty scaling factor
ϵ_m	convergence threshold for merit
ϵ_u	convergence threshold for control variable
ϵ_c	constraint satisfaction threshold
C_t	needle pose at step t
N	number of trajectory steps
b	needle insertion distance at each step
ξ_t	twist applied at step t
ζ_t	needle deviation at step t
$\bar{\zeta}$	max. needle deviation
l_g	needle length free for grasping
l_n	needle length
κ_n	needle natural curvature
ϵ_{entry}	entry position tolerance
ϵ_{exit}	exit position tolerance

Schulman et al. [107] proposed a nonconvex sequential optimization for bevel-tip needle steering based on a curvature-constrained kinematic model. Sen et al. [83] extended this concept for curved needles. Building on this model, we formulate the stitching needle trajectory generation problem as a nonconvex constrained optimization problem to be solved by ℓ_1 penalized trust-region based sequential optimization.

5.2.1 Sequential convex programming

The goal of a general nonlinear programming problem (NLP) is to find an optimal set of parameters ($u \in \mathbb{R}^n$) that minimizes a given cost function ($f_{(u)}$), while satisfying a set of equality ($h_{(u)}$) and inequality ($g_{(u)}$) constraints as

$$\begin{aligned} &\arg\min_{u} f_{(u)} \\ &\text{subject to} \\ &\qquad h_{i_{(u)}} = 0 \quad i = 0, \cdots, n_{eq} \\ &\qquad g_{j_{(u)}} \leq 0 \quad j = 0, \cdots, n_{ineq}. \end{aligned} \tag{5.1}$$

In general, NLPs are difficult to solve, but one effective method for trajectory optimization problems is the use of Sequential Convex Programming (SCP) [108],[107]. The SCP method approximates a given NLP with convex subproblems (convex cost function, convex inequality constraints, and affine equality constraints) and sequentially updates the solution estimation by solving the convex subproblem. NLP convexification can be performed by representing the cost function and constraints through Taylor series expansions around the current iterate u as

$$\begin{aligned} \tilde{f}_{(\Delta u)} &= f_{(u)} + \nabla f_{(u)}^{\top} \Delta u + \frac{1}{2} \Delta u^{\top} \nabla^2 f_{(u)} \Delta u \\ \tilde{h}_{i_{(\Delta u)}} &= h_{i_{(u)}} + \nabla h_{i_{(u)}}^{\top} \Delta u \\ \tilde{g}_{j_{(\Delta u)}} &= g_{j_{(u)}} + \nabla g_{j_{(u)}}^{\top} \Delta u. \end{aligned} \tag{5.2}$$

The solution of the subproblem generates a step Δu to update the current solution towards the optimal solution. However, infeasibility is a common problem during an SCP that can be caused by a poor convexification or initial solution guess. One possible solution is to move the infeasible constraints into the objective function as penalties, which will converge to zero and ensure feasibility. Compared with other common penalty functions, the use of ℓ_1 penalty provides an exact representation of the deviation from the constraint and numerical stability [109]. To avoid complications because of its non-smooth characteristic, it is common to implement ℓ_1 penalties by using slack variables (v, w and t in Equation 5.3) that do not modify the convexity of the original problem [110].

A trust-region method is also used to ensure global convergence. It is represented as a step boundary constraint $\|\Delta u\|_p \leq \delta$, where δ is the trust region radius and p represents the p-norm used for the boundary conditions. The minimization problem is reformulated for a positive penalty μ as:

$$\arg \min_{\Delta u, v, w, t} \tilde{f}_{(\Delta u)} + \mu \left[\sum_{i=1}^{n_{eq}} (v_i + w_i) + \sum_{j=1}^{n_{ineq}} t_j \right]$$

subject to

$$\begin{aligned} \tilde{h}_{i_{(\Delta u)}} &= v_i - w_i \,, \quad i = 0, \cdots, n_{eq} \\ \tilde{g}_{j_{(\Delta u)}} &\leq t_i \,, \quad j = 0, \cdots, n_{ineq} \\ v, w, t &\geq 0 \\ \|\Delta u\|_\infty &\leq \delta. \end{aligned} \tag{5.3}$$

To determine the acceptance of the current step, the SCP method uses the actual-to-predicted cost reduction ratio ρ, defined by

$$\rho = \frac{\phi_{1_{(u,\mu)}} - \phi_{1_{(u+\Delta u,\mu)}}}{q_{(0)} - q_{(\Delta u)}} \tag{5.4}$$

where the ℓ_1 merit function $\phi_{1_{(u,\mu)}}$ is given as

$$\phi_{1_{(u,\mu)}} \coloneqq f_{(u)} + \mu \left[\sum_{i=0}^{n_{eq}} |h_{i_{(u)}}| + \sum_{j=0}^{n_{ineq}} [g_{j_{(u)}}]^+ \right] \tag{5.5}$$

and the approximation of the objective function $q_{(\Delta u,\mu)}$ is given as

$$\begin{aligned} q_{(\Delta u,\mu)} \coloneqq & f_{(u)} + \nabla f_{(u)}^\top \Delta u + \frac{1}{2} \Delta u^\top H_{(u)} \Delta u + \\ & \mu \left[\sum_{i=0}^{n_{eq}} |h_{i_{(u)}} + \nabla h_{i_{(u)}}^\top \Delta u| + \sum_{j=0}^{n_{ineq}} [g_{j_{(u)}} + \nabla g_{j_{(u)}}^\top \Delta u]^+ \right] \end{aligned} \tag{5.6}$$

where $H_{(u)}$ represents a numerical approximation of the Hessian of the cost function $\nabla^2 f_{(u)}$, and $[y]^+ = \max(y, 0)$. If $q_{(\Delta u,\mu)}$ is an accurate representation of the ℓ_1 merit function ($\rho \geq \eta_a$), the trust region δ is expanded by a factor of γ_e and the control variable Δu is updated. On the other hand, if δ is rejected ($\rho \leq \eta_r$), δ is shrunk by a factor of γ_s. The sequential iteration stops when the convergence criteria are satisfied or the maximum number of iterations is exceeded. The outline of ℓ_1 penalized trust-region based sequential optimization algorithm is shown in Algorithm 1.

Algorithm 1: ℓ_1 penalty trust region based sequential optimization

Input: η_r,η_a,γ_s, γ_e, μ_s, ϵ_m, ϵ_u,ϵ_c, f, g, h

for $l \coloneqq 0, \cdots, l_{max}$ **do**

  for $k \coloneqq 0, \cdots, k_{max}$ **do**

    Define the local approximated cost and constraint functions:

    $\tilde{f}^{(k)}_{(\Delta u)} = f_{(u^{(k)})} + \nabla f^{\top}_{(u^{(k)})}\Delta u + \frac{1}{2}\Delta u^{\top} H^{(k)} \Delta u$

    $\tilde{g}^{(k)}_{(\Delta u)} = g_{(u^{(k)})} + \nabla g^{\top}_{(u^{(k)})}\Delta u$

    $\tilde{h}^{(k)}_{(\Delta u)} = h_{(u^{(k)})} + \nabla h^{\top}_{(u^{(k)})}\Delta u$

    for $m \coloneqq 0, \cdots, m_{max}$ **do**

      Formulate smooth quadratic programming problem including slack variables

      $\Delta u \leftarrow \arg\min_{\Delta u, v, w, t} \tilde{f}^{(k)}_{(\Delta u)} + \mu^{(l)} \left[\sum_{i=1}^{n_{eq}}(v_i + w_i) + \sum_{j=1}^{n_{ineq}} t_j\right]$

      Determine the actual-to-predicted cost ratio:

      $\rho = \frac{\phi^{(k)}_{1_{(u,\mu)}} - \phi^{(k)}_{1_{(u+\Delta u,\mu)}}}{q^{(k)}_{(0)} - q^{(k)}_{(\Delta u)}}$

      Determine step acceptance and update trust region radius

      if $\rho \leq \eta_r$ **then**

        $\delta^{(m+1)} = \gamma_s \delta^{(m)}$ shrink trust region

      else if $\eta_r \leq \rho \leq \eta_a$ **then**

        $u^{(k+1)} = u^{(k)} + \Delta u$

        break

      else

        $u^{(k+1)} = u^{(k)} + \Delta u$

        $\delta^{(m+1)} = \gamma_e \delta^{(m)}$ expand trust region

        break

      end

    end

    if $\phi^{(k)}_{1_{(u,\mu)}} - \phi^{(k+1)}_{1_{(u,\mu)}} \leq \epsilon_m$ **then**

      break

  end

  Verify largest constraint violation and update

  if $\max([g_{(u^{(k+1)})}]^{+}, |h_{(u^{(k+1)})}|)$ **then**

    $u_{optimal} \leftarrow u^{(k+1)}$

    break

  else

    $u^{(l+1)} = \mu_s u^{(l)}$

  end

end

5.2.2 Problem definition

The stitching needle trajectory is discretized into N time intervals and represented as a set of needle tip poses at each time step $C : \{C_0, C_1, ..., C_{N-1}\}$, where each pose is defined as a homogeneous transformation matrix $C_t = \begin{bmatrix} R_t & p_t \\ \mathbf{0} & 1 \end{bmatrix} \in SE(3)$. The stitching trajectory planning problem can be defined as:

Input: The desired entry and exit points $p_{entry}, p_{exit} \in \mathbb{R}^3$ are given by the surgeon during the calibration process. The needle size and shape (l_n, κ_n), the suture depth z_{depth}, and the maximum needle curvature allowed $\bar{\zeta}$ are also provided.

Output: The set of desired needle tip poses $C_d : \{C_{d_0}, C_{d_1}, \cdots, C_{d_{N-1}}\}$.

Assumptions: It is assumed that the gripper holds the needle firmly so that needle deviation during the stitching task does not occur. We consider a flexible tissue where the friction forces produced during the tissue penetration can be neglected. This assumption allows us to ignore the dynamic effects of the interaction between the needle and tissue, and model the needle kinematics as a rigid link extension of the arm robot.

5.2.3 Optimization model

The stitching needle trajectory generation problem is formulated as a constrained optimization problem as:

$$\arg\min_{\Delta u} \alpha_1 Cost_L + \alpha_2 Cost_{O_{entry}} + \alpha_3 Cost_{O_{exit}} \tag{5.7}$$

subject to:

$$\log\left(C_{t+1}(\exp(\xi_t)C_t)^{-1}\right)^{\vee} = \mathbf{0}_6 \tag{5.8}$$

$$(p_0 - p_{entry})^{\top}(p_0 - p_{entry}) \leq \epsilon_{entry}^2 \tag{5.9}$$

$$(p_{N-1} - p_{exit})^{\top}(p_{N-1} - p_{exit}) \leq \epsilon_{exit}^2 \tag{5.10}$$

$$Nb + l_g \leq l_n \tag{5.11}$$

$$C_{N/2} \leq z_{depth} \tag{5.12}$$

$$\zeta_t - \bar{\zeta} \leq 0\,, \quad t = 0, \cdots, N-1 \tag{5.13}$$

$$-\zeta_t - \bar{\zeta} \leq 0\,, \quad t = 0, \cdots, N-1 \tag{5.14}$$

$$\zeta_{t+1}^2 - \zeta_t^2 \leq 0\,, \quad t = 0, \cdots, N-2 \tag{5.15}$$

Costs (Eqn. 5.7): The objective of $Cost_L$ penalizes longer trajectories to reduce tissue trauma. Additionally, needle deviation from an orthogonal pose with respect to the tissue surface is also penalized during the needle insertion with $Cost_{O_{entry}}$ and extraction $Cost_{O_{exit}}$ to facilitate tissue penetration. The concrete forms of these cost functions are given as

$$Cost_L \coloneqq Nb \tag{5.16}$$

$$Cost_{O_{entry}} \coloneqq \log(R_0 R_{entry}^\top)^\vee \tag{5.17}$$

$$Cost_{O_{exit}} \coloneqq \log(R_{N-1} R_{exit}^\top)^\vee. \tag{5.18}$$

Stitching kinematic constraints (Eqn. 5.8): We use the curvature constrained kinematic model proposed by Sen et al. [83] for curved needles. At each time step, the needle is inserted following a fixed length b with a curvature $\kappa_n + \zeta_t$, where κ_n is the needle natural curvature and ζ_t are local curvature changes applied for needle reorientation at the time step t. The transformation between consecutive needle poses can be represented as a twist ${}^{n_{tip}}\xi_t \coloneqq [0, 0, b, -b(\kappa_n + \zeta_t), 0, 0]$, composed of a translational motion b along the needle tip tangent vector, and a rotational motion $-b(\kappa_n + \zeta_t)$ along the normal vector to the needle's plane. The corresponding twist in the base frame can then be computed as $\xi_t = Ad_{n_{tip}}\,{}^{n_{tip}}\xi_t$, with the adjoint matrix defined as $Ad_{n_{tip}} = \begin{bmatrix} R_{n_{tip}} & [p_{n_{tip}}]_\times R_{n_{tip}} \\ \mathbf{0} & R_{n_{tip}} \end{bmatrix}$.

The lie algebra exponential and logarithmic mapping in $SE(3)$ [111] are used to represent the kinematic constraint $C_{t+1} = \exp(\xi_t) C_t$ at each time step.

Desired entry/exit port constraints (Eqn. 5.9–5.10): The generated trajectory entry/exit points should be within a tolerance $\epsilon_{entry}, \epsilon_{exit}$ from the desired entry/exit point, respectively.

Needle length constraints (Eqn. 5.11): The total trajectory length including the distance for the forceps to hold the needle $Nb + l_g$ should be less than the needle length l_n.

Suture depth constraint (Eqn. 5.12): The trajectory should have a minimum depth z_{depth} to ensure a proper tissue penetration.

Needle reorientation constraints (Eqn. 5.13–5.15): Small curvature changes ζ_t are allowed from the needle natural curvature κ_n , but bounded by $\bar{\zeta}$ and limited to be monotonically decreasing.

5.3 Constrained motion planning

EES constrains the motion of the surgical tools on the nostril and along the nasal cavity. It is important to ensure safe tool manipulation by implementing a remote center of motion constraint over the nostril. We propose a constrained motion planning based on a guidance virtual fixture along the optimally generated needle trajectory and a concurrent constrained inverse kinematic solver implementation based on a task-priority IK method and a nonlinear programming IK method. Figure 5.5 depicts a block diagram of the proposed constrained motion control. Given a surgeon's pose command through the serial-link interface X_h, the scaled desired end-effector pose X_{des} is computed. A guidance virtual fixture constrains the needle tip motion along a continuous optimal trajectory and generates the constrained end-effector pose X_c. Then, a concurrent IK solver performs a simultaneous prioritized Jacobian-based inverse kinematics and a constrained nonlinear optimization inverse kinematics. The task-priority IK can find a solution faster, but joint limits and higher priority tasks may produce convergence failures. On the other hand, nonlinear optimization takes longer to compute, but can better handle joint limits and additional constraints. When one of them finds an acceptable solution, the other IK solver is stopped, and the computed joint command q_d is distributed between the robot arm ($q_{d_{arm}}$) and the articulated forceps ($q_{d_{forceps}}$).

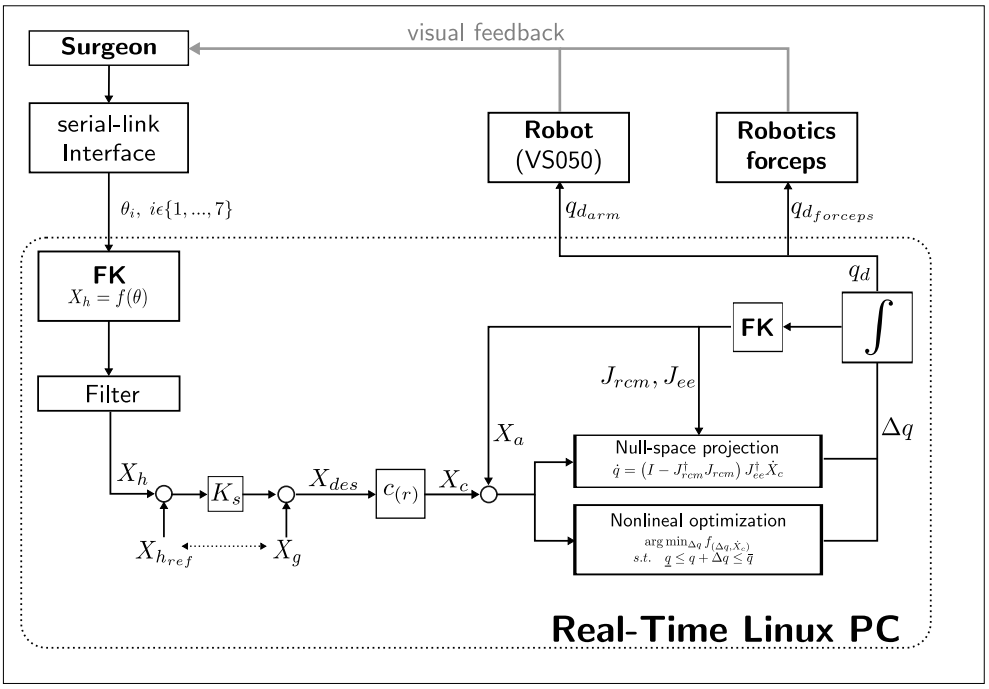

Figure 5.5: Block diagram of the proposed constrained motion planning.

5.3.1 Guidance Virtual Fixture

A continuous and differentiable position and orientation trajectory $c_{(s)} = \{c_{p_{(s)}}, c_{o_{(s)}}\}$ is generated by cubic b-spline interpolation of the control points from the optimal stitching needle trajectory obtained in Section 5.2 (see Figure 5.6) as

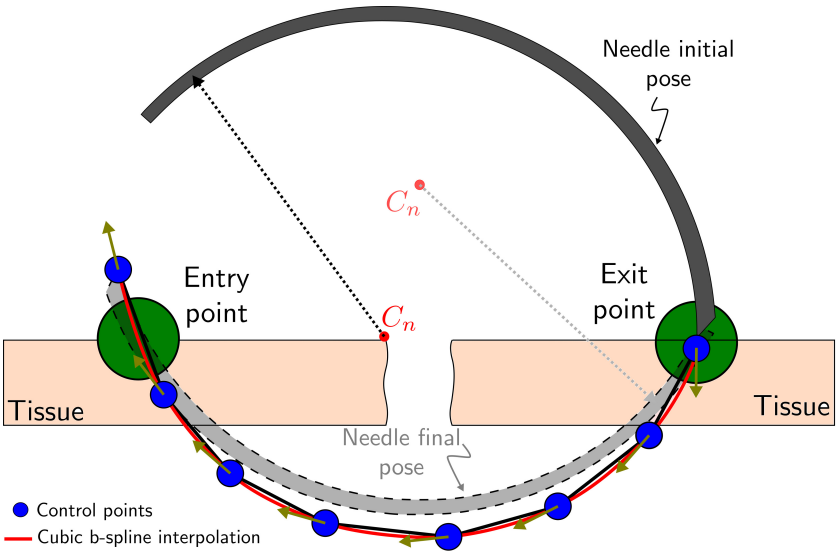

Figure 5.6: Guidance virtual fixture.

$$
\begin{aligned}
c_{p_{(s)}} &= p_{d_0} + \sum_{i=0}^{N-1} (p_{d_i} - p_{d_0}) \mathbf{B}_{i,k+1_{(s)}}, \quad s \in [0,1] \\
c_{o_{(s)}} &= \exp\left(\sum_{i=0}^{N-1} log(R_{d_i} R_{d_0}^{\top})^{\vee} \mathbf{B}_{i,k+1_{(s)}} \right) R_{d_0}, \quad s \in [0,1]
\end{aligned}
\tag{5.19}
$$

$$
\begin{aligned}
\mathbf{B}_{i,1_{(s)}} &= \begin{cases} 1 & \lambda_i \leq t < \lambda_{i+1} \\ 0 & \text{otherwise} \end{cases} \\
\mathbf{B}_{i,k+1_{(s)}} &= \frac{s - \lambda_i}{\lambda_{i+k} - \lambda_i} \mathbf{B}_{i,k_{(s)}} + \frac{\lambda_{i+k+1} - s}{\lambda_{i+k+1} - \lambda_{i+1}} \mathbf{B}_{i+1,k_{(s)}}
\end{aligned}
\tag{5.20}
$$

where $\mathbf{B}_{i,k+1_{(s)}}$ are the b-spline basis functions and $\lambda_i \in [0,1]$ are uniformly distributed knots.

The continuous trajectory $c_{(s)}$ is then used as reference curve for a hard virtual fixture [101]. The desired position p_{des} obtained from the user interface command is projected onto the normalized tangent direction of the reference curve $u_{c_{(s_t)}} = \frac{\delta_{c_{(s_t)}}}{\|\delta_{c_{(s_t)}}\|}$, with $\delta_{c_{(s_t)}} = \frac{d}{ds} c_{p_{(s)}}|_{s=s_t}$, and scaled by K_c to update the interpolation parameter s_t as:

$$
s_{t+1} = s_t + K_c \Delta {p_{des}}^{\top} u_{c_{(s_t)}}. \tag{5.21}
$$

The constrained pose X_c is then computed by evaluating $c_{(s_{t+1})}$.

5.3.2 Task-priority inverse kinematics

The RCM constraint can be defined as a higher priority task in a hierarchical null-space projection method [112]. Based on this, Azimian et al. [113] proposed an RCM implementation by projecting the end-effector tracking task over the null space of the RCM task. Sandoval et al. [114] proposed a similar RCM implementation for a torque-controlled redundant manipulator, but based on the RCM distance to a trocar that does not depend on the insertion velocity. We define the RCM constraint problem also using a prioritized Jacobian method but defining the RCM position as the nostril projection onto the forceps shaft.

Figure 5.7 depicts the variables used in the RCM Jacobian computation. The unit vector $\hat{p_s} = \frac{p_s}{\|p_s\|}$, with $p_s = p_{j_8} - p_{j_7}$ represents the direction of the forceps shaft and $p_r = p_{nostril} - p_{j_7}$ is the vector between the nostril and the base of the forceps shaft. We can compute the RCM position by

$$p_{rcm} = p_{j_7} + p_r^\top \hat{p}_s \hat{p}_s. \tag{5.22}$$

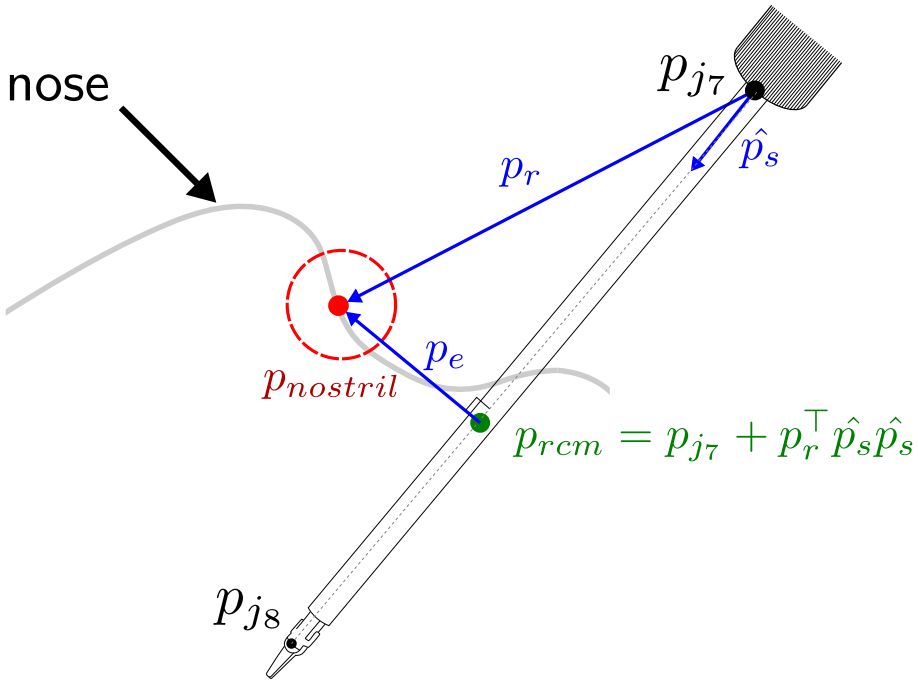

Figure 5.7: Remote center of motion characterization.

Then we can differentiate p_{rcm}:

$$\frac{\partial p_{rcm}}{\partial q} = J_{j_7} + \hat{p}_s \frac{\partial p_r^\top \hat{p}_s}{\partial q} + p_r^\top \hat{p}_s \frac{\partial \hat{p}_s}{\partial q}. \tag{5.23}$$

By matrix differentiation, we obtain the RCM Jacobian J_{rcm} as:

$$J_{rcm} = \frac{\partial p_{rcm}}{\partial q} = \left(\mathbf{I} - \hat{p}_s \hat{p}_s{}^\top\right) J_{j_7} + \left(\hat{p}_s p_r^\top + p_r^\top \hat{p}_s \mathbf{I}\right) \frac{\partial \hat{p}_s}{\partial q} \tag{5.24}$$

where

$$\frac{\partial \hat{p}_s}{\partial q} = \frac{1}{\|p_s\|} \left(\mathbf{I} - \hat{p}_s \hat{p}_s{}^\top\right) \left(J_{j_8} - J_{j_7}\right). \tag{5.25}$$

We assume the constrained motion planning is initialized with the forceps shaft over the nostril, so that the RCM Jacobian velocity must be zero $\dot{p}_{rcm} = 0$. By projecting the end-effector tracking task over the null space of the RCM task, we can compute the joint velocity command as:

$$\dot{q} = \left(\mathbf{I} - J_{rcm}^\dagger J_{rcm}\right) J_{ee}^\dagger \dot{e}_{ee} \tag{5.26}$$

5.3.3 Nonlinear optimization inverse kinematics

The previous prioritized Jacobian-based IK can be susceptible to convergence failure when close to joint limits. Beeson et al. [104] proposed a concurrent inverse kinematics library for generic inverse kinematic (TRAC-IK) that addresses the convergence problem by simultaneously executing a nonlinear optimization-based IK solver. Similarly, we propose a constrained nonlinear optimization problem to be executed simultaneously with our previous task-priority IK solver.

We define the constrained nonlinear optimization problem as:

$$\begin{aligned} \arg\min_{\Delta q} \quad & \beta_1 e_p^\top e_p + \beta_2 e_o^\top e_o + \beta_3 e_{rcm}^\top e_{rcm} + \beta_4 \Delta q^\top \Delta q \\ s.t. \quad & \underline{q} \leq q + \Delta q \leq \bar{q} \end{aligned} \tag{5.27}$$

where β_n are the positive coefficients, $e_{p_{(\Delta q)}} = p_{des} - p_{(\Delta q)}$ is the Cartesian position end-effector error, $e_{o_{(\Delta q)}} = \log\left(R_{des} R_{(\Delta q)}^\top\right)^\vee$ is the orientation error expressed using the logarithmic mapping, and $e_{rcm_{(\Delta q)}} = p_{nostril} - p_{rcm_{(\Delta q)}}$ is the RCM position error. By increasing the value of β_3 over the other coefficients, we can prioritize the minimization of an RCM error.

5.4 Experiments and discussion

5.4.1 Implementation details

The proposed methodology is implemented on a 2.4 GHz Core i7 CONTEC computer running Linux (Ubuntu 16.04, Canonical) with real-time patches (RT-PREEMPT) and the Robot Operating System (ROS) framework on top of it. The control loop runs at a rate of 125 Hz in synchronous mode with the industrial robot arm controller. In each cycle, the current robot pose is updated, and then, the target robot pose is computed and sent to the robot arm and the articulated forceps controllers [106]. The rigid body kinematic is implemented by using the template library Pinocchio [115], and the optimization framework is built on CasADi [116], which allows C-code generation to speed up the optimization process. The quadratic programming solver used for the stitching needle trajectory generation is the Gurobi Optimizer [117]. For the nonlinear optimization-based IK, we use the IPOPT solver [118]. The weighting factors α, and β from the objective functions in Equation 5.7 and Equation 5.27 were obtained experimentally.

5.4.2 Simulation environment

To evaluate the performance of our proposed scheme, we initially validate the trajectory generation and constrained motion control in a simulation environment developed with CoppeliaSim EDU V4.1 [119], in which we model the nostril constraint and simulate the robot kinematics. Figure 5.8a shows the simulation environment created for system implementation. In Figure 5.8b, the desired entry point and exit points are represented with a blue and red sphere respectively, with a 1.5 mm tolerance radius. An initial needle trajectory is provided as a half-turn trajectory that follows the natural needle curvature and is shown as a red curve. The blue curve corresponds to the optimal trajectory obtained from the sequential convex optimization, and the green curve corresponds to the forceps tip reference trajectory. In simulation conditions, the stitching needle trajectory generation takes approximately 500ms and the maximum RCM deviation is 1mm.

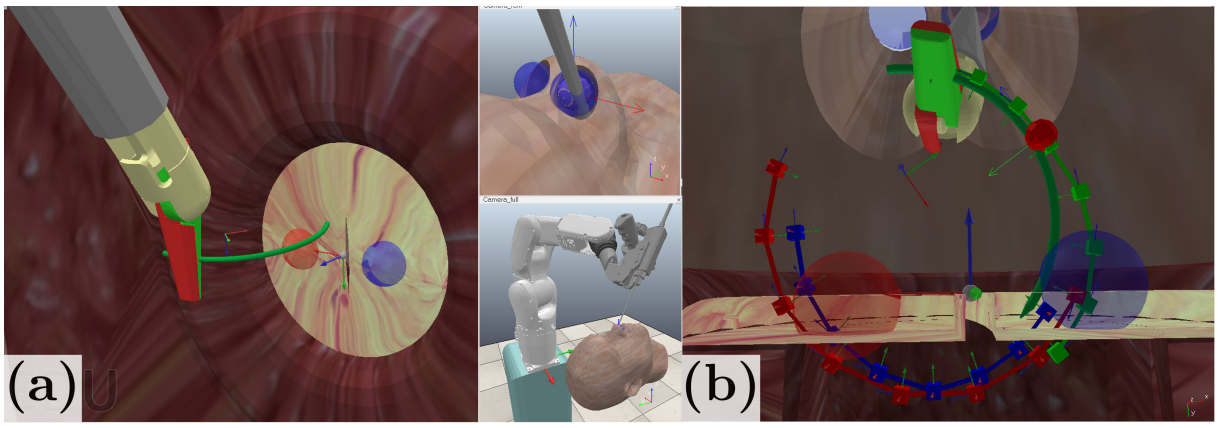

Figure 5.8: (**a**) CoppeliaSim simulation environment. (**b**) Trajectory generated in simulation.

5.4.3 Stitching experiments

We conduct a stitching experiment to validate the performance of our system in a physically realistic environment and compare three stitching modes:

- **Manual**: by using conventional forceps for endoscopic endonasal surgery.
- **Robot-assisted**: by using the proposed method. The system generates the optimal trajectory and constrains the needle pose.
- **Autonomous**: the robot starts in a fixed initial pose and executes the stitching task without human assistance.

The experimental setup is presented in Figure 5.9.

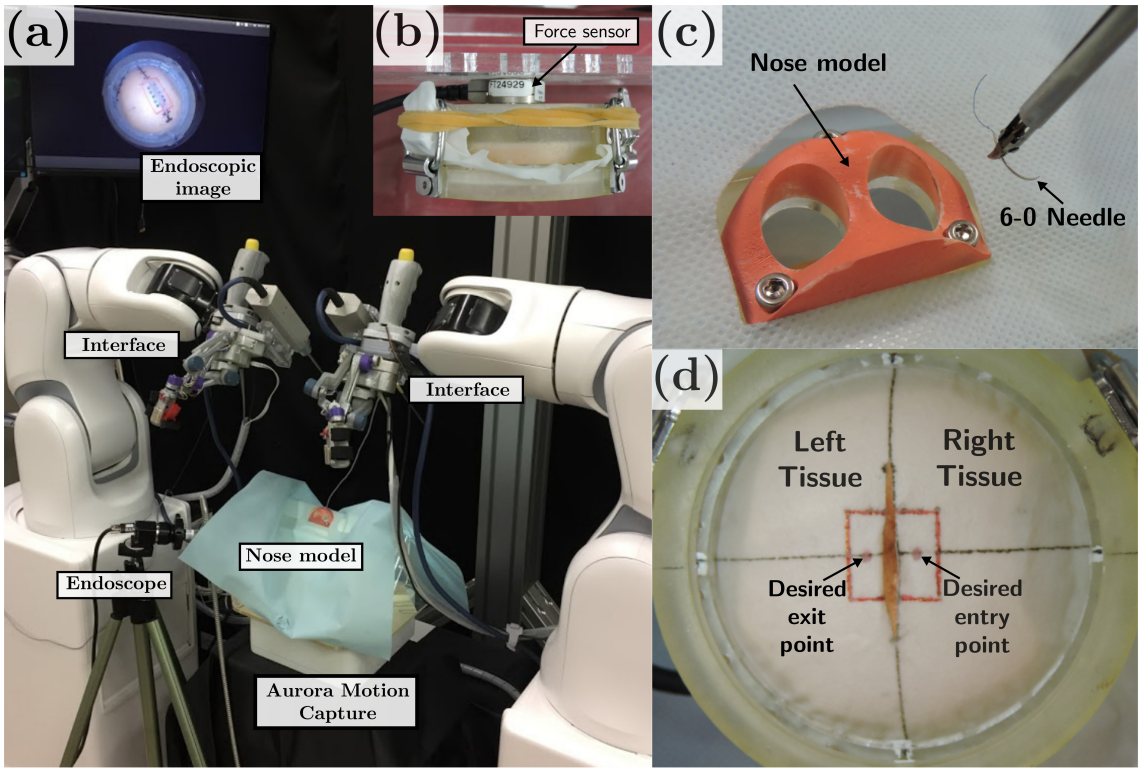

Figure 5.9: (**a**) Experiment setup. (**b**) Force sensor placed behind the stitching testbed. (**c**) 3D printed nose model. (**d**) Stitching testbed.

It comprises a 3D printed nose model and an acrylic platform for the stitching testbed. A six-axis force/torque sensor (Nano17, ATI Industrial Automation, North Carolina, USA) is placed behind the elastic tissue to measure the compression and shear forces applied to the tissue during the stitching task (see Figure 5.9b). We assumed the compression forces to be perpendicular to the tissue surface and aligned to the z-axis of the force sensor, whereas the shear forces are tangential to the tissue surface. For the force analysis, we use the root sum square (RSS) of the resulting forces given by the square root of the sum of squares of the forces in each direction measured by the force sensor (Eq. 5.28). The forces are sampled at a frequency of 1 kHz and the testbed's weight related forces are subtracted.

$$\text{RSS Force} = \sqrt{f_x^2 + f_y^2 + f_z^2} \tag{5.28}$$

A magnetic motion capture system (Aurora, Northern Digital Inc., Ontario, Canada) is used to record the motions of the surgical instrument for both manual and robot operation.

A rigid endoscope (30°, 2.7 mm diameter) is used for visualization and targeting the area of interest.

The participants were asked to perform a stitching task. The task consisted of grasping a 6-0 surgical needle and puncturing an elastic tissue, on which the desired entry/exit point was marked (see Figure 5.9d). In the *manual* stitching mode, the participants were asked to follow a three-step procedure: (1) puncture the right tissue from the marked point. (2) penetrate the right tissue until the needle tip is placed under the left tissue, and (3) puncture the left tissue from underneath. In the *robot-assisted* stitching mode, the participants were asked to position the needle in a suitable initial position and command the needle insertion speed (0–0.5mm/s) and direction through the user interface, whilst the robot constrained the needle pose along the reference trajectory and kept the RCM constraints. In both cases, assistance from the surgical tool on the left hand was also allowed to tighten the left tissue, but no needle manipulation assistance was allowed. The *autonomous* stitching mode was performed without human assistance, and the needle insertion speed was set at the maximum allowed (0.5mm/s). In all stitching modes, the trial starts with the needle grasped by the surgical tool and placed about 2cm above the center frame of the tissue.

Seven subjects between the ages of 25 and 35 who had no previous surgical training took part in this experiment. Before the experiment, each participant was instructed about the procedure and practiced for up to 10 minutes until they were familiarized with the operation. All subjects were given informed consent before they participated in the study. This study was approved by the Ethical Research Committee of Nagoya University.

We evaluate participants' performance using the following metrics:

- **Task completion time (s)**: the total time in which participants performed the task.

$$\text{Completion Time} \coloneqq t_{end} - t_{start} \tag{5.29}$$

- **Success ratio (%)**: the percentage of succeed stitching from the total number of attempts.

$$\text{Success ratio} \coloneqq \frac{\text{number of success insertions}}{\text{number of attempts}} \tag{5.30}$$

- **Entry point error (mm)**: the Euclidean distance between the desired entry point and the actual entry point.

$$\text{Entry point error} \coloneqq \|p_{entry_{des}} - p_{entry_{act}}\| \tag{5.31}$$

- **Exit point error (mm)**: the Euclidean distance between the desired exit point and the actual exit point.

$$\text{Exit point error} \coloneqq \|p_{exit_{des}} - p_{exit_{act}}\| \tag{5.32}$$

- **Maximum RCM error (mm)**: the maximum Euclidean distance between the RCM position and the center of the nostril.

$$\text{Max. RCM error} \coloneqq \max\left(\|p_{rcm} - p_{nostril}\|\right) \ . \tag{5.33}$$

- **Maximum force (N)**: the maximum force applied on the tissue.

$$\text{Maximum force} \coloneqq \max(|f|) \tag{5.34}$$

- **Distribution of force samples**: graphical representation of the number of force measurements within equally distributed force intervals with respect to the total number of force samples obtained during the stitching task.

Figure 5.10 summarizes the results obtained for the three stitching modes (*manual*, *robot-assisted*, and *autonomous*). The number of trials and success ratio per subject are presented in Tables 5.2–5.4. Note that the task completion time, entry point error, and exit point error are computed only for the successful trials.

Table 5.2: Success ratio for Manual Operation

Subject	**# of Trials**	**# of Success**	**Success Ratio (%)**
1	12	4	33.3
2	15	4	26.7
3	12	4	33.3
4	8	2	25.0
5	15	2	13.3
6	12	1	8.3
7	8	3	37.5
Total	82	20	
Mean Success Ratio (%)			25.4

Table 5.3: Success ratio for Robot-assisted Operation

Subject	**# of Trials**	**# of Success**	**Success Ratio (%)**
1	12	5	41.7
2	10	5	50.0
3	8	5	62.5
4	8	5	62.5
5	7	4	57.1
6	6	6	100.0
7	6	5	83.3
Total	57	35	
Mean Success Ratio (%)			65.3

Table 5.4: Success ratio for Autonomous operation

# of Trials	**# of Success**	**Success Ratio (%)**
16	7	43.8

Figure 5.10a depicts the task completion time. The reliability of using the robotic system is evident as the standard deviation is greatly reduced compared with the manual operation. The median time for the robot-assisted is about 25% longer than that of the autonomous mode. However, there was no significant improvement with respect to the manual operation. Figure 5.10b shows the success ratio of the stitching task, where the robot-assisted mode achieved the success ratio of 65.3%, much higher than that of the autonomous mode (43.8%) and the manual operation (25.4%). Needle-tissue interaction forces arising at the needle tip may induce undesired changes in the needle orientation, which could result in a failed attempt to penetrate the tissue. Unlike the autonomous mode where the speed is fixed, the robot-assisted mode allows the user to freely control the insertion speed and reduce the effect of such interaction forces. Moreover, in the robot-assisted mode, the user can also retract the needle along the optimal trajectory if proper penetration is not achieved (e.g., because of tissue deformation). The needle can be pulled back and reinserted without the need of recomputing the optimal trajectory as long as the needle orientation with respect to the forceps tip has not changed. As a consequence, the robot-assisted mode can require additional time to complete the task compared with the autonomous method.

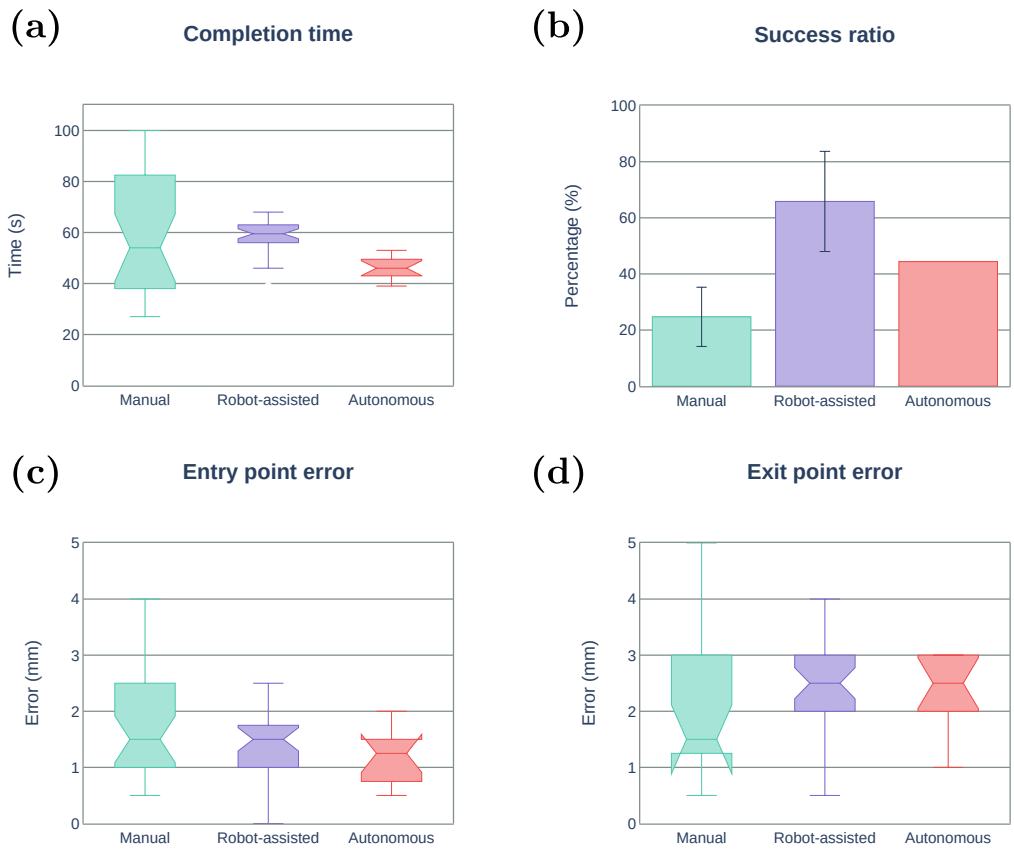

Figure 5.10: Experimental results of the stitching task.

Therefore, a trade-off exists between assuring proper tissue penetration and the stitching completion time. Figure 5.10c-d depicts the entry/exit point error. The autonomous mode showed the smaller entry point error among the three cases, but no improvement in exit point error can be observed when using the robotic system. This is expected, as the stitching trajectory generation algorithm will try to keep the changes in the curvature bounded to facilitate tissue penetration and reduce tissue deformation, which can produce a larger deviation from the desired exit point. By increasing the allowed range of curvature changes, the exit error is expected to be reduced, but higher forces might be generated.

We evaluate the constrained motion planning performance by the maximum RCM error defined as the minimum distance between the nostril and the forceps shaft. The RCM constraint implemented for the robotic operation (including robot-assisted and autonomous modes) was able to keep the maximum RCM error below 4.2 mm as shown in Figure 5.11a. Figure 5.11b illustrates an example of the forceps displacement from the center of the nostril during task execution by one of the participants. The blue line represents the position of the forceps shaft intersection with the nostril plane for the manual operation, and the red

line represents the intersection trajectory obtained when using the robotic system.

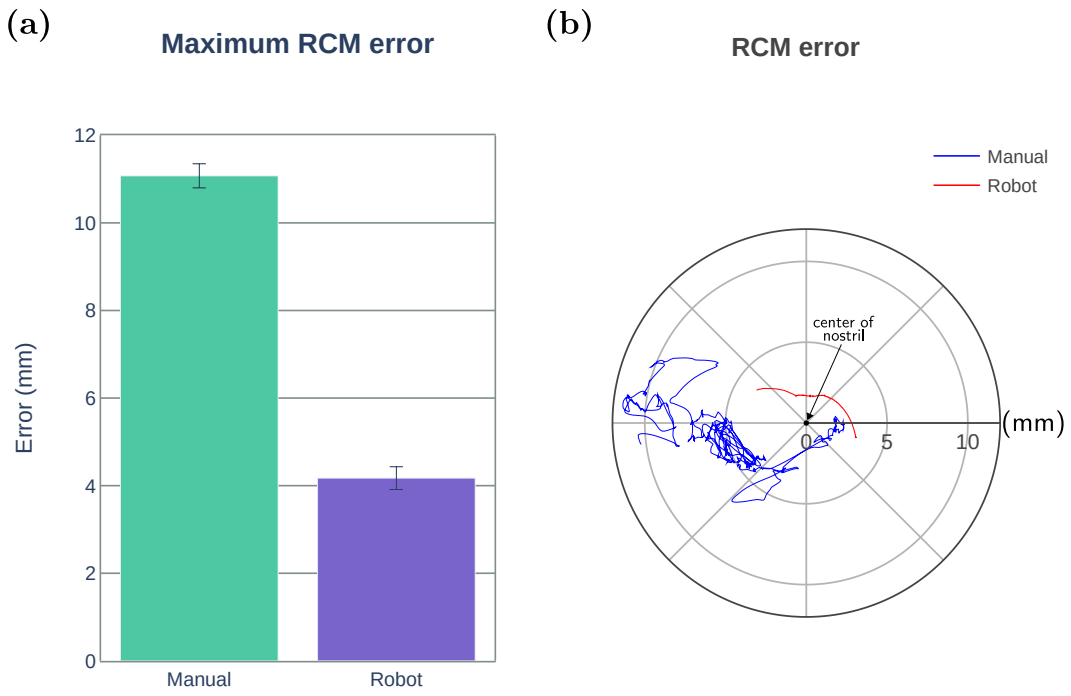

Figure 5.11: Comparison of the constrained motion planning performance.

We also measured the forces applied during the stitching task to evaluate the possibility of potential tissue trauma. Table 5.5 summarizes the maximum forces applied. The use of the robotic system demonstrates a reduction of approximately 70% of the maximum force compared with manual operation. During the manual operation, frequent undesired collision of the forceps with the tissue occurred because of the lack of depth information by the 2D endoscopic visualization. Figure 5.12 shows the distribution of the force samples contained within the force intervals of 0.005 N. The larger the number of bins close to 0 N, the smaller amount of continuous force applied to the tissue (less potential damage). With the use of the robot system (robot-assisted and autonomous modes), approximately 90% of the force samples are within the range of [0,0.05] N, while for the manual operation, 90% of the samples are found within the range of [0,0.15] N. The use of the robotic system reduced effectively the range of forces applied over the tissue in approximately 66%.

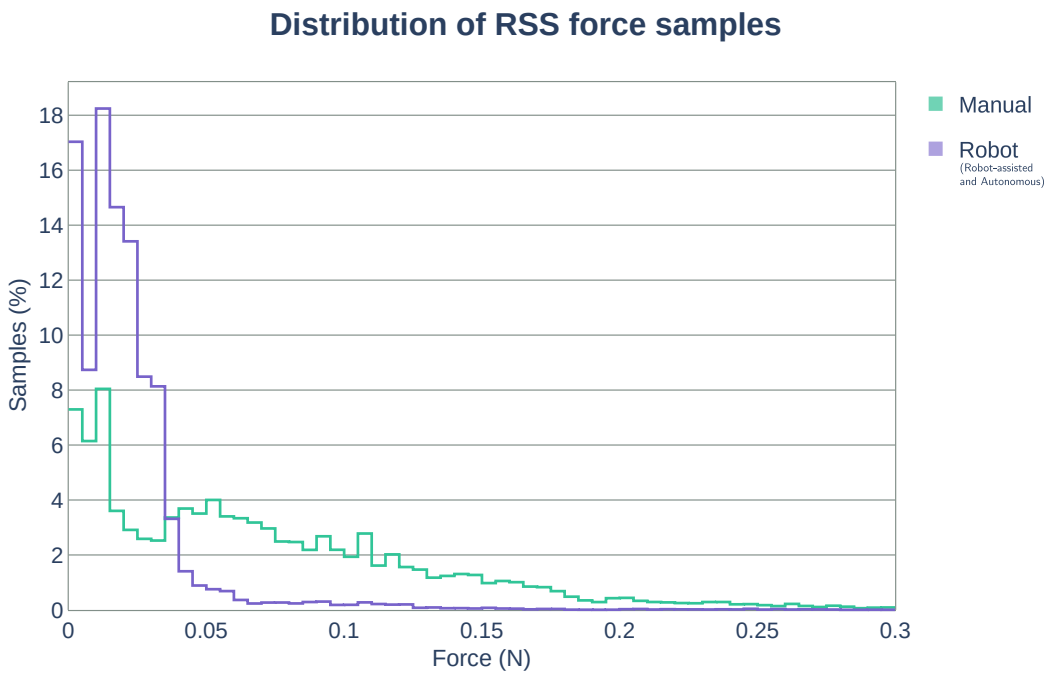

Figure 5.12: Distribution of RSS force samples. In robot operation (robot-assisted and autonomous modes), approximately 90% of the RSS force samples are within the range of [0,0.05] N. For the manual operation, the 90% of the samples are within the range of [0,0.15] N.

Table 5.5: Maximum force measured during the stitching task

	Mean (N)	**SD (N)**
Manual operation	0.349	0.178
Robotic system	0.096	0.046

Figure 5.13a depicts the needle positioned into the stitching workspace. An example of the complete stitching sequence achieved with the robot-assisted mode is shown in Figure 5.13b-g, with a total duration of 58s. The user initially places the needle close to the entry point (t=0, Figure 5.13b) and activates the robot-assisted stitching sequence. An optimal trajectory is generated with an initial needle tip pose. The robot reorients the needle (Figure 5.13c) and approaches to the entry point located over the tissue (Figure 5.13d). Under the user control, the needle penetrates the right tissue (Figure 5.13e), commands the needle under the tissue toward the exit point (Figure 5.13f), and finally penetrates the left tissue from underneath (Figure 5.13g).

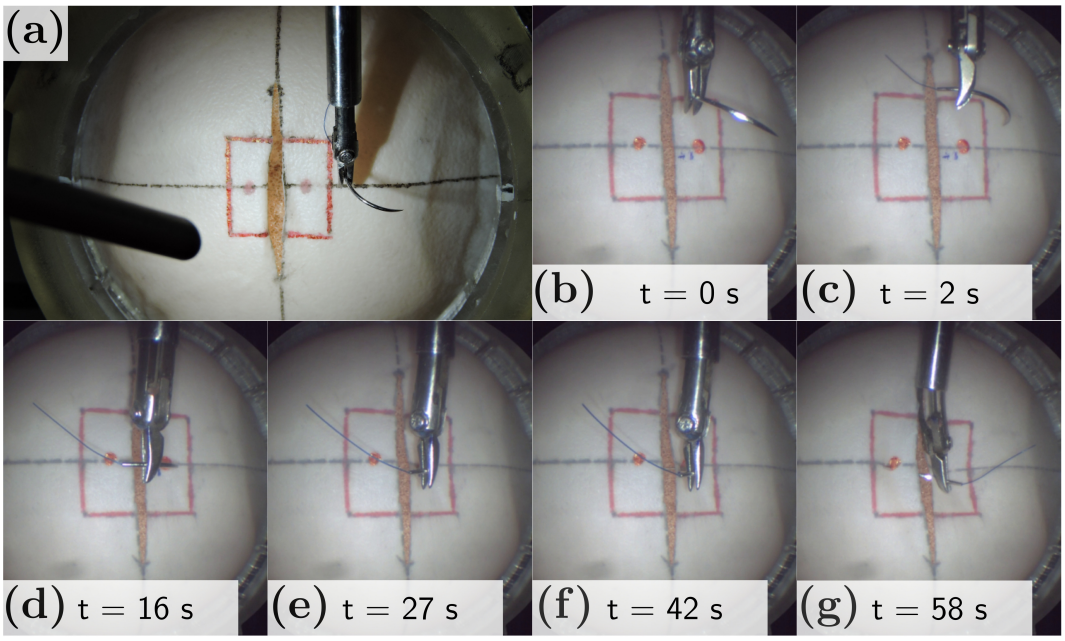

Figure 5.13: (**a**) Needle grasped by the 4-DOF forceps before starting the stitching task. (**b-g**) Robot-assisted stitching task sequence. (b) Initial needle positioning about 2cm over the tissue. (c) Needle reorientation. (d) Needle approach to the tissue. (e) Right tissue penetration. (f) Needle insertion. (g) Left tissue penetration from underneath.

Chapter 6

Conclusion

Endoscopic endonasal surgery remains a challenging task for robotic-assisted surgical systems. Few previous works have addressed the limitations of neurosurgical applications, but highly complex surgical tasks remain unsolved. This thesis is intended to provide an innovative framework and strategy to tackle the issues associated with robotic-assisted EES.

6.1 Contributions

A first contribution of this thesis is to introduce a cooperative human-robot interface for robot-assisted endoscopic endonasal surgery. We evaluated several design concepts to enhance intuitiveness and safety during highly dexterous anatomically constrained surgical tasks in EES. We finally selected a hybrid interface composed of a force-based and a serial-link interface. The surgical task was divided into stages, each of which was concerned with specific challenges and constraints.

The force-based interface was chosen for positioning, insertion, and extraction of the articulated forceps because of the simplicity and natural control arising from human-robot physical interaction. During the positioning stage, the robot worked in gravity compensation mode and followed the force applied by the surgeon over the interface. Then, the robot constrains the forceps motion along a virtual remote-center-of-motion fixed in the nostril during the insertion and extraction states.

Once the tool was placed at the target workspace, we used the serial-link interface for precise control of the articulated forceps while keeping a virtual remote-center-of-motion at the nostril. Workspace constraints were implemented based on virtual fixtures to limit the tool workspace and ensure safe motion inside the patient. We defined the constraints from

a phantom head to determine the sinus cavity and the dura workspace. We proposed a state-based real-time controller that combined an admittance controller for positioning and insertion/extraction with a position tracker to control the forceps tip during the surgical task, ensuring a smooth transition between these stages.

Finally, we compared the performance of our proposed system with the use of conventional surgical instruments in common surgical tasks. A reachability experiment using a phantom head showed the capability of reaching the areas of interest such as the pituitary area in a smooth and precise way compared to the use of conventional instruments. Pick-and-place and block-in-hole experiments were performed to evaluate the manipulation of objects inside the nasal cavity under the anatomic constraints imposed by the nostril. The results showed a noticeable improvement in the motion smoothness. Besides, a needle stitching task was conducted to test the system in the highly dexterous surgical task. The robotic system showed a similar time for each phase, with a high reduction in the Regrasping phase, where passing the needle was challenging with the requirement of keeping a proper needle orientation. The same experiment showed a reduction in the force distribution applied over the tissue during the needle insertion/extraction. The results demonstrated that the proposed system could control and safely constrain the motion of bimanual robot arms with articulated multi-DOF forceps for endoscopic endonasal surgical tasks.

The second contribution of this thesis is the developing of an optimization-based trajectory generation method for robot-assisted stitching based on sequential convex optimization to find the optimal needle trajectory, and a dual concurrent IK solver to constrain the surgical tool around the nostril. The trajectory generation algorithm was capable of generating an optimal trajectory subject to the suturing requirements. The surgeon can regenerate the trajectory online through the user interface. Integrating two constrained IK methods allowed us to achieve the convergence of the solution and meet time constraints.

We compared the performance of our system with a manual stitching operation and an autonomous operation. The results showed a noticeable improvement in the stitching success ratio and reduction of the force interaction with the tissue. In the robot-assisted mode, the surgeon can control the initial positioning and insertion speed, which avoids potential rotations in the needle and ensure a proper initial tissue penetration. However, the task completion time did not significantly improve compared with the manual mode. Although the robot task completion time could be reduced by increasing the maximum speed allowed for the needle, it might reduce the insertion success. In addition, the interaction force distribution showed that the use of robot reduced about 70% of the forces

shown in a manual stitching task. This is important to avoid any potential tissue trauma. The results also showed that the proposed system could safely constrain the motion of the articulated multi-DOF forceps for an endoscopic endonasal surgical stitching.

6.2 Remaining issues

The human-robot interface proposed was designed to target specific tasks in endonasal surgery. However, it can be extended to general applications by increasing the interface workspace range. Control of additional surgical tool functions other than grasping (e.g., cutting, drilling) should be explored for transparent integration with the proposed machine state scheme, which could require the addition of extra activation buttons or remapping of the current interface configuration. The addition of haptic feedback should also be considered to enhance the surgeon's perception and keep a safety force interaction with the surrounded tissues.

For the semi-autonomous motion implementation, only a stitching task has been studied. For a full suturing task, additional steps, such as knot tying and needle passing, must also be included. We characterized the stitching task with a simplified kinematic model without considering tissue and needle deformation. The dual concurrent inverse kinematic framework implemented to ensure solution convergence in the constrained trajectory generation considered three main constraints: joint limits, joint velocity, and remote-center-of-motion. However, to keep a safe needle-tissue interaction, additional features could also be used such as manipulability and motion smoothness. Experimental validations of the proposed system have only been performed in a phantom environment. A real environment includes dynamic changes to be considered, such as; blood, camera occlusion, tissue texture and elasticity, among others. In a real surgical setup, interaction with other instruments and equipment are frequently needed, and emergency protocols should also be considered. Further experimentation would be required to establish a clear performance evaluation of the proposed system.

6.3 Future directions

The future research direction is dedicated to solving the issues mentioned in the previous section:

- Characterization of additional surgical tasks such as knot tying for optimization-based constrained trajectory generation that tracks the user interface motion during

a complete multi-throw suturing sequence.

- Use of endoscope visual feedback for dynamic generation of virtual constraints in semi-autonomous robot execution and to provide relevant information to the surgeon through augmented reality. Additional factors such as light, tool collision, and tissue deformation should also be included.
- Transfer learning from surgeons' demonstrations and artificial intelligence techniques could be used to provide enhanced autonomy and decision-making.
- Experimental evaluation of the proposed system, including subjects with prior surgical training in a realistic environment.

Bibliography

[1] M. R. Drange, N. R. Fram, V. Herman-Bonert, and S. Melmed, "Pituitary Tumor Registry: A Novel Clinical Resource1," *J. Clin. Endocrinol. Metab.*, vol. 85, no. 1, pp. 168–174, 2000.

[2] Surgery for pituitary tumors. American Cancer Society. Accesed on: Dec. 20, 2020. [Online]. Available: https://www.cancer.org/cancer/pituitary-tumors/treating/surgery.html

[3] J. K. Liu, K. Das, M. H. Weiss, E. R. Laws, and W. T. Couldwell, "The history and evolution of transsphenoidal surgery," *J. Neurosurg.*, vol. 95, no. 6, pp. 1083–1096, 2001.

[4] R. Jankowski, J. Auque, C. Simon, J. C. Marchal, H. Hepner, and M. Wayoff, "How i do it: Head and neck and plastic surgery: Endoscopic pituitary tumor surgery," *Laryngoscope*, vol. 102, no. 2, pp. 198–202, 1992.

[5] A. R. Cutler, G. Barkhoudarian, C. F. Griffiths, and D. F. Kelly, "Transsphenoidal endoscopic skull base surgery: state of the art and future perspective," *Innov. Neurosurg.*, vol. 1, no. 1, pp. 15–35, 2013.

[6] M. Messerer, J. C. D. battista, G. Raverot, S. Kassis, J. Dubourg, V. Lapras, J. Trouillas, G. Perrin, and E. Jouanneau, "Evidence of improved surgical outcome following endoscopy for nonfunctioning pituitary adenoma removal: Personal experience and review of the literature," *Neurosurg. Focus*, vol. 30, no. 4, p. E11, 2011.

[7] Y. Esquenazi, W. I. Essayed, H. Singh, E. Mauer, M. Ahmed, P. J. Christos, and T. H. Schwartz, "Endoscopic endonasal versus microscopic transsphenoidal surgery for recurrent and/or residual pituitary adenomas," *World Neurosurgery*, vol. 101, pp. 186–195, 2017.

[8] K. Kenan, A. İhsan, O. Dilek, C. Burak, K. Gurkan, and C. Savas, "The learning curve in endoscopic pituitary surgery and our experience," *Neurosurg. Rev.*, vol. 29, no. 4, pp. 298–305, 2006.

[9] Y. Kawanishi, Y. Fujimoto, N. Kumagai, M. Takemura, M. Nonaka, E. Nakai, N. Masahira, T. Nakajo, and K. Shimizu, "Evaluation of two- and three-dimensional visualization for endoscopic endonasal surgery using a novel stereoendoscopic system in a novice: a comparison on a dry laboratory model," *Acta Neurochirurgica*, vol. 155, no. 9, pp. 1621–1627, 2013.

[10] A. Garcia-Ruiz, M. Gagner, J. H. Miller, C. P. Steiner, and J. F. Hahn, "Manual vs Robotically Assisted Laparoscopic Surgery in the Performance of Basic Manipulation and Suturing Tasks," *Arch. Surg.*, vol. 133, no. 9, pp. 957–961, 1998.

[11] J. P. Ruurda, I. A. M. J. Broeders, B. Pulles, F. M. Kappelhof, and C. v. d. Werken, "Manual robot assisted endoscopic suturing: Time-action analysis in an experimental model," *Surg. Endosc.*, vol. 18, no. 8, pp. 1249–1252, 2004.

[12] O. Alimoglu, J. Sagiroglu, I. Atak, A. Kilic, T. Eren, M. Caliskan, and G. Bas, "Robot-assisted laparoscopic (ral) procedures in general surgery," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 12, no. 3, pp. 427–430, 2016.

[13] M. Kranzfelder, C. Staub, A. Fiolka, A. Schneider, S. Gillen, D. Wilhelm, H. Friess, A. Knoll, and H. Feussner, "Toward increased autonomy in the surgical OR: needs, requests, and expectations," *Surg. Endosc.*, vol. 27, no. 5, pp. 1681–1688, 2013.

[14] R. H. Taylor, "Robots as surgical assistants: Where we are, wither we are tending, and how to get there," in *Proc. Artificial Intelligence in Medicine*, 1997, pp. 1–11.

[15] P. Schleer, S. Drobinsky, M. d. l. Fuente, and K. Radermacher, "Toward versatile cooperative surgical robotics: a review and future challenges," *Int. J. Copmut. Assit. Radiol. Surg.*, pp. 1–14, 2019.

[16] N. Nathoo, M. C. Çavuşoğlu, M. A. Vogelbaum, and G. H. Barnett, "In Touch with Robotics: Neurosurgery for the Future," *Neurosurgery*, vol. 56, no. 3, pp. 421–433, 2005.

[17] R. Taylor, B. Mittelstadt, H. Paul, W. Hanson, P. Kazanzides, J. Zuhars, B. Williamson, B. Musits, E. Glassman, and W. Bargar, "An image-directed robotic system for precise orthopaedic surgery," *IEEE Trans. Robot. Automat.*, vol. 10, no. 3, pp. 261–275, 1994.

[18] W. L. Bargar, A. Bauer, and M. Börner, "Primary and Revision Total Hip Replacement Using the Robodoc System," *Clinical Orthopaedics and Related Research*, vol. 354, pp. 82–91, 1998.

[19] M. H. L. Liow, P. L. Chin, H. N. Pang, D. K.-J. Tay, and S.-J. Yeo, "THINK surgical TSolution-One (Robodoc) total knee arthroplasty," *SICOT-J*, vol. 3, p. 63, 2017.

[20] P. Cinquin, E. Bainville, C. Barbe, E. Bittar, V. Bouchard, L. Bricault, G. Champleboux, M. Chenin, L. Chevalier, Y. Delnondedieu, L. Desbat, V. Dessenne, A. Hamadeh, D. Henry, N. Laieb, S. Lavallee, J. M. Lefebvre, F. Leitner, Y. Menguy, F. Padieu, O. Peria, A. Poyet, M. Promayon, S. Rouault, P. Sautot, J. Troccaz, and P. Vassal, "Computer assisted medical interventions," *IEEE Eng. Med. Biol. Mag.*, vol. 14, no. 3, pp. 254–263, 1995.

[21] G. S. Guthart and J. K. Salisbury, "The Intuitive/sup TM/ telesurgery system: overview and application," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1, 2000, pp. 618–621.

[22] S. DiMaio, M. Hanuschik, and U. Kreaden, "The da vinci surgical system," in *Surgical Robotics: Systems Applications and Visions*, J. Rosen, B. Hannaford, and R. M. Satava, Eds. Springer US, 2011, pp. 199–217.

[23] T. H. Massie and J. K. Salisbury, "The PHANTOM Haptic Interface: A Device for Probing Virtual Objects," in *Proc. Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 1994.

[24] A. Tobergte, P. Helmer, U. Hagn, P. Rouiller, S. Thielmann, S. Grange, A. Albu-Schaffer, F. Conti, and G. Hirzinger, "The sigma.7 haptic interface for MiroSurge: A new bi-manual surgical console," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2011, pp. 3023–3030.

[25] A. Simorov, S. R. Otte, C. M. Kopietz, and D. Oleynikov, "Review of surgical robotics user interface: what is the best way to control robotic surgery?" *Surgical Endoscopy*, vol. 26, no. 8, pp. 2117–2125, 2012.

[26] M. Piccigallo, F. Focacci, O. Tonet, G. Megali, C. Quaglia, and P. Dario, "Hand-held robotic instrument for dextrous laparoscopic interventions," *Int. J. Med. Robot.*, vol. 4, no. 4, pp. 331–338, 2008.

[27] A. H. Zahraee, J. K. Paik, J. Szewczyk, and G. Morel, "Toward the Development of a Hand-Held Surgical Robot for Laparoscopy," *IEEE/ASME Trans. Mechatron.*, vol. 15, no. 6, pp. 853–861, 2010.

[28] R. A. MacLachlan, B. C. Becker, J. C. Tabarés, G. W. Podnar, L. A. Lobes, Jr., and C. N. Riviere, "Micron: An Actively Stabilized Handheld Tool for Microsurgery," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 195–212, 2012.

[29] C. Song, D. Y. Park, P. L. Gehlbach, S. J. Park, and J. U. Kang, "Fiber-optic OCT sensor guided "SMART" micro-forceps for microsurgery," *Biomed. Opt. Express*, vol. 4, no. 7, p. 1045, 2013.

[30] C. J. Payne and G.-Z. Yang, "Hand-Held Medical Robots," *Ann. Biomed. Eng.*, vol. 42, no. 8, pp. 1594–1605, 2014.

[31] M. Jinno, T. Sunaoshi, and S. Omori, "Manipulator and control method therefor," U.S. Patent US8 237 388B2, 2012.

[32] M. S. Hallbeck, D. Oleynikov, K. Done, T. Judkins, A. DiMartino, J. Morse, and L. N. Verner, "Ergonomic handle and articulating laparoscopic tool," U.S. Patent US8 585 734B2, 2013.

[33] H. Yamashita, A. Iimura, E. Aoki, and S. T. o. Japan ..., "Development of endoscopic forceps manipulator using multi-slider linkage mechanisms," in *Proc. The 1st Asian Symposium on Computer Aided Surgery-Robotic and Image guided Surgery*, 2005.

[34] A. Hassan-Zahraee, B. Herman, and J. Szewczyk, "Mechatronic Design of a Hand-Held Instrument with Active Trocar for Laparoscopy," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1, 2011, pp. 1890–1895.

[35] M. Jakopec, S. J. Harris, F. R. y. Baena, P. Gomes, and B. L. Davies, "The Acrobot system for total knee replacement," *Ind. Robot.*, vol. 30, no. 1, pp. 61–66, 2003.

[36] B. Hagag, R. Abovitz, H. Kang, B. Schmitz, and M. Conditt, "Rio: Robotic-arm interactive orthopedic system makoplasty: User interactive haptic orthopedic robotics," in *Surgical Robotics: Systems Applications and Visions*, J. Rosen, B. Hannaford, and R. M. Satava, Eds. Springer US, 2011, pp. 219–246.

[37] R. Taylor, P. Jensen, L. Whitcomb, A. Barnes, R. Kumar, D. Stoianovici, P. Gupta, Z. Wang, E. Dejuan, and L. Kavoussi, "A Steady-Hand Robotic System for Microsurgical Augmentation," *Int. J. Robot. Res.*, vol. 18, no. 12, pp. 1201–1210, 1999.

[38] O. Schneider, J. Troccaz, O. Chavanon, and D. Blin, "PADyC: a synergistic robot for cardiac puncturing," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 3, 2000, pp. 2883–2888.

[39] Y. Zhan, X. Duan, and J. Li, "Review of comanipulation robot in surgery," in *2015 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2015, pp. 1466–1471.

[40] J. S. Schneider, J. Burgner, R. J. Webster, and P. T. Russell, "Robotic surgery for the sinuses and skull base: what are the possibilities and what are the obstacles?" *Curr. Opin. Otolaryngol. Head Neck Surg.*, vol. 21, no. 1, pp. 11–6, 2013.

[41] J. Y. K. Lee, J. B. W. O'Malley, J. G. Newman, G. S. Weinstein, B. Lega, J. Diaz, and M. S. Grady, "Transoral Robotic Surgery of the Skull Base: A Cadaver and Feasibility Study," *ORL*, vol. 72, no. 4, pp. 181–187, 2010.

[42] E. Y. Hanna, C. Holsinger, F. DeMonte, and M. Kupferman, "Robotic Endoscopic Surgery of the Skull Base: A Novel Surgical Approach," *Arch. Otolaryngol. Head. Neck. Surg.*, vol. 133, no. 12, p. 1209, 2007.

[43] J. Burgner, D. C. Rucker, H. B. Gilbert, P. J. Swaney, P. T. Russell, K. D. Weaver, and R. J. Webster, "A Telerobotic System for Transnasal Surgery," *IEEE/ASME Trans. Mechatron.*, vol. 19, no. 3, pp. 996–1006, 2014.

[44] T. A. Travaglini, P. J. Swaney, K. D. Weaver, and R. J. W. III, "Initial Experiments with the Leap Motion as a User Interface in Robotic Endonasal Surgery," *Mech. Mach. Theory*, vol. 37, pp. 171–179, 2015.

[45] Á. M. García, I. Rivas, J. P. Turiel, V. Muñoz, J. C. F. Marinero, E. de la Fuente, and J. M. Sabater, "Integration of a surgical robotic co-worker in an endoscopic neurosurgical assistance platform," in *Robot 2019: Fourth Iberian Robotics Conference*, M. F. Silva, J. Luís Lima, L. P. Reis, A. Sanfeliu, and D. Tardioli, Eds., 2020, pp. 453–464.

[46] C.-A. Nathan, V. Chakradeo, K. Malhotra, H. D'Agostino, and R. Patwardhan, "The Voice-Controlled Robotic Assist Scope Holder AESOP for the Endoscopic Approach to the Sella," *Skull Base*, vol. 16, no. 03, pp. 123–131, 2006.

[47] M. Matinfar, C. Baird, A. Batouli, R. Clatterbuck, and P. Kazanzides, "Robot-Assisted Skull Base Surgery," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2007, pp. 865–870.

[48] T. Xia, C. Baird, G. Jallo, K. Hayes, N. Nakajima, N. Hata, and P. Kazanzides, "An integrated system for planning, navigation and robotic assistance for skull base surgery," *Int. J. Med. Robot.*, vol. 4, no. 4, pp. 321–330, 2008.

[49] Y. He, Y. Hu, P. Zhang, B. Zhao, X. Qi, and J. Zhang, "Human–Robot Cooperative Control Based on Virtual Fixture in Robot-Assisted Endoscopic Sinus Surgery," *Appl. Sci.*, vol. 9, no. 8, p. 1659, 2019.

[50] N. Hogan, "Impedance Control: An Approach to Manipulation: Part I," *J. Dyn. Syst. Meas. Control*, vol. 107, no. 1, 1985.

[51] K. Kosuge and N. Kazamura, "Control of a robot handling an object in cooperation with a human," in *Proc. IEEE International Workshop on Robot and Human Communication*, 1997, pp. 142–147.

[52] T. Tsumugiwa, R. Yokogawa, and K. Hara, "Variable Impedance Control with Virtual Stiffness for Human-Robot Cooperative Peg-in-Hole Task," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, vol. 2, 2002, pp. 1075–1081.

[53] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion Learning and Adaptive Impedance for Robot Control during Physical Interaction with Humans," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 4326–4332.

[54] V. Duchaine and C. M. Gosselin, "General Model of Human-Robot Cooperation Using a Novel Velocity Based Variable Impedance Control," in *Proc. EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2007, pp. 1–6.

[55] R. Ikeura and H. Inooka, "Variable impedance control of a robot for cooperation with a human," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 3, 1995, pp. 3097–3102 vol.3.

[56] P. Evrard and A. Kheddar, "Homotopy Switching Model for Dyad Haptic Interaction in Physical Collaborative Tasks," in *Proc. EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, vol. 1, 2009, pp. 45–50.

[57] A. Gams, B. Nemec, A. J. Ijspeert, and A. Ude, "Coupling Movement Primitives: Interaction with the Environment and Bimanual Tasks," *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 816–830, 2014.

[58] P. Donner and M. Buss, "Cooperative Swinging of Complex Pendulum-Like Objects: Experimental Evaluation," *IEEE Trans. Robot.*, vol. 32, no. 3, pp. 744–753, 2016.

[59] S. Ikemoto, H. B. Amor, T. Minato, B. Jung, and H. Ishiguro, "Physical Human-Robot Interaction: Mutual Learning and Adaptation," *IEEE Robot. Automat. Mag.*, vol. 19, no. 4, pp. 24–35, 2012.

[60] L. Peternel and J. Babič, "Learning of compliant human–robot interaction using full-body haptic interface," *Adv. Robot.*, vol. 27, no. 13, pp. 1003–1012, 2013.

[61] J. Krüger, T. Lien, and A. Verl, "Cooperation of human and machines in assembly lines," *CIRP Ann. Manuf. Technol.*, vol. 58, no. 2, pp. 628–646, 2009.

[62] H. Rininsland, "ARTEMIS. A telemanipulator for cardiac surgery," *J. Cardiothorac. Surg.*, vol. 16, no. Supplement_2, pp. S106–S111, 1999.

[63] Y. Kwoh, J. Hou, E. Jonckheere, and S. Hayati, "A robot with improved absolute positioning accuracy for CT guided stereotactic brain surgery," *IEEE Trans. Biomed. Eng.*, vol. 35, no. 2, pp. 153–160, 1988.

[64] S. Lavallee, J. Troccaz, L. Gaborit, P. Cinquin, A. Benabid, and D. Hoffmann, "Image guided operating robot: a clinical application in stereotactic neurosurgery," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1, 1992, pp. 618–624.

[65] V. Zanotto, P. Boscariol, A. Gasparetto, A. Lanzutti, R. Vidoni, N. D. Lorenzo, P. Gallina, A. D. Via, and A. Rossi, "A master-slave haptic system for neurosurgery," *Appl. Bionics Biomech.*, vol. 8, no. 2, pp. 209–220, 2011.

[66] C. Ho, E. Tsakonas, K. Tran, K. Cimon, M. Severn, M. Mierzwinski-Urban, J. Corcos, and S. Pautler, "Robot-Assisted Surgery Compared with Open Surgery and Laparoscopic Surgery: Clinical Effectiveness and Economic Analyses," *CADTH Technology Report*, 2011.

[67] A. Bettini, S. Lang, A. Okamura, and G. Hager, "Vision Assisted Control for Manipulation using Virtual Fixtures: Experiments at Macro and Micro Scales," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 4, 2002, pp. 3354–3361.

[68] J. Funda, R. Taylor, B. Eldridge, S. Gomory, and K. Gruben, "Constrained Cartesian motion control for teleoperated surgical robots," *IEEE Trans. Robot. Automat.*, vol. 12, no. 3, pp. 453–465, 1996.

[69] A. Kapoor, M. Li, and R. H. Taylor, "Spatial Motion Constraints for Robot Assisted Suturing Using Virtual Fixtures," in *Proc. Medical Image Computing and Computer-Assisted Intervention International Conference*, 2005, pp. 89–96.

[70] L. Rosenberg, "Virtual fixtures: Perceptual tools for telerobotic manipulation," in *Proc. IEEE Virtual Reality Annual International Symposium*, 1993, pp. 76–82.

[71] L. Joly and C. Andriot, "Imposing motion constraints to a force reflecting telerobot through real-time simulation of a virtual mechanism," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1, 1995, pp. 357–362 vol.1.

[72] A. Knoll, H. Mayer, C. Staub, and R. Bauernschmitt, "Selective automation and skill transfer in medical robotics: a demonstration on surgical knot-tying," *Int. J. Med. Robot.*, vol. 8, no. 4, pp. 384–397, 2012.

[73] T. Osa, N. Sugita, and M. Mitsuishi, "Online Trajectory Planning in Dynamic Environments for Surgical Task Automation," in *Proc. Robot. Sci. Syst.*, 2014.

[74] J. v. d. Berg, S. Miller, D. Duckworth, H. Hu, A. Wan, X.-Y. Fu, K. Goldberg, and P. Abbeel, "Superhuman Performance of Surgical Tasks by Robots using Iterative Learning from Human-Guided Demonstrations," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2010, pp. 2074–2081.

[75] H. Kang and J. Wen, "Autonomous suturing using minimally invasive surgical robots," in *Proc. IEEE Int. Conf. Control Autom.*, 2000, pp. 742–747.

[76] D.-L. Chow and W. Newman, "Trajectory Optimization of Robotic Suturing," in *Proc. IEEE Int. Conf. Tech. Prac. Robot. Appl.*, 2015, pp. 1–6.

[77] F. Nageotte, P. Zanne, C. Doignon, and M. d. Mathelin, "Stitching Planning in Laparoscopic Surgery: Towards Robot-assisted Suturing," *Int. J. Robot. Res.*, vol. 28, no. 10, pp. 1303–1321, 2009.

[78] T. Liu and M. C. Cavusoglu, "Needle Grasp and Entry Port Selection for Automatic Execution of Suturing Tasks in Robotic Minimally Invasive Surgery," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 552–563, 2016.

[79] C. Staub, T. Osa, A. Knoll, and R. Bauernschmitt, "Automation of Tissue Piercing using Circular Needles and Vision Guidance for Computer Aided Laparoscopic Surgery," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2010, pp. 4585–4590.

[80] C. D'Ettorre, G. Dwyer, X. Du, F. Chadebecq, F. Vasconcelos, E. D. Momi, and D. Stoyanov, "Automated pick-up of suturing needles for robotic surgical assistance," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 1370–1377.

[81] S. Iyer, T. Looi, and J. Drake, "A Single Arm, Single Camera System for Automated Suturing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 239–244.

[82] S. A. Pedram, P. Ferguson, J. Ma, E. Dutson, and J. Rosen, "Autonomous Suturing via Surgical Robot: an Algorithm for Optimal Selection of Needle Diameter, Shape, and Path," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 2391–2398.

[83] S. Sen, A. Garg, D. V. Gealy, S. McKinley, Y. Jen, and K. Goldberg, "Automating Multi-Throw Multilateral Surgical Suturing with a Mechanical Needle Guide and Sequential Convex Optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 4178–4185.

[84] R. C. Jackson and M. C. Cavusoglu, "Needle Path Planning for Autonomous Robotic Surgical Suturing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 1669–1675.

[85] K. Watanabe, T. Kanno, K. Ito, and K. Kawashima, "Single-Master Dual-Slave Surgical Robot With Automated Relay of Suture Needle," *IEEE Trans. Ind. Electron.*, vol. 65, no. 8, pp. 6343–6351, 2018.

[86] L. Adhami and E. Coste-Maniere, "Optimal planning for minimally invasive surgical robots," *IEEE Trans. Robot. Automat.*, vol. 19, no. 5, pp. 854–863, 2003.

[87] B. Kehoe, G. Kahn, J. Mahler, J. Kim, A. Lee, A. Lee, K. Nakagawa, S. Patil, W. D. Boyd, P. Abbeel, and K. Goldberg, "Autonomous Multilateral Debridement with the Raven Surgical Robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 1432–1439.

[88] A. Murali, S. Sen, B. Kehoe, A. Garz, S. McFarland, S. Patil, W. D. Boyd, S. Lim, P. Abbeel, and K. Goldberg, "Learning by Observation for Surgical Subtasks: Multilateral Cutting of 3D Viscoelastic and 2D Orthotropic Tissue Phantoms," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 1202–1209.

[89] B. Thananjeyan, A. Garg, S. Krishnan, C. Chen, L. Miller, and K. Goldberg, "Multilateral Surgical Pattern Cutting in 2D Orthotropic Gauze with Deep Reinforcement Learning Policies for Tensioning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 2371–2378.

[90] S. Javdani, S. Tandon, J. Tang, J. F. O'Brien, and P. Abbeel, "Modeling and Perception of Deformable One-Dimensional Objects," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 1607–1614.

[91] R. C. Jackson, R. Yuan, D.-L. Chow, W. S. Newman, and M. C. avuolu, "Real-Time Visual Tracking of Dynamic Surgical Suture Threads," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 3, pp. 1078–1090, 2018.

[92] N. Padoy and G. D. Hager, "Human-Machine Collaborative Surgery Using Learned Models," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 5285–5292.

[93] K. B. Reed, A. Majewicz, V. Kallem, R. Alterovitz, K. Goldberg, N. J. Cowan, and A. M. Okamura, "Robot-assisted needle steering," *IEEE Robot. Automat. Mag.*, vol. 18, no. 4, pp. 35–46, 2011.

[94] Z. Chen, A. Malpani, P. Chalasani, A. Deguet, S. S. Vedula, P. Kazanzides, and R. H. Taylor, "Virtual Fixture Assistance for Needle Passing and Knot Tying," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2016, pp. 2343–2350.

[95] M. Selvaggio, A. M. G. E, R. Moccia, F. Ficuciello, and B. Siciliano, "Haptic-guided shared control for needle grasping optimization in minimally invasive robotic surgery," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2019, pp. 3617–3623.

[96] M. M. Marinho, H. Ishida, K. Harada, K. Deie, and M. Mitsuishi, "Virtual Fixture Assistance for Suturing in Robot-Aided Pediatric Endoscopic Surgery," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 524–531, 2019.

[97] G. A. Fontanelli, G.-Z. Yang, and B. Siciliano, "A Comparison of Assistive Methods for Suturing in MIRS," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2018, pp. 4389–4395.

[98] M. M. Marinho, A. Nakazawa, J. Nakanishi, T. Ueyama, Y. Hasegawa, J. Arata, M. Mitsuishi, and K. Harada, "Conceptual Design of a Versatile Robot for Minimally Invasive Transnasal Microsurgery," in *Proc. Int. Sym. Micro-Nano Mech. Hum. Sci.*, 2016, pp. 1–3.

[99] J. Arata, Y. Fujisawa, R. Nakadate, K. Kiguchi, K. Harada, M. Mitsuishi, and M. Hashizume, "Compliant four degree-of-freedom manipulator with locally deformable elastic elements for minimally invasive surgery," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 2663–2669.

[100] A. Q. Keemink, H. v. d. Kooij, and A. H. Stienen, "Admittance control for physical human–robot interaction," *Int. J. Robot. Res.*, vol. 37, no. 11, pp. 1421–1444, 2018.

[101] A. Bettini, P. Marayong, S. Lang, A. M. Okamura, and G. D. Hager, "Vision-Assisted Control for Manipulation Using Virtual Fixtures," *IEEE Trans. Robot.*, vol. 20, no. 6, pp. 953–966, 2004.

[102] M. M. Marinho, M. C. Bernardes, and A. P. L. Bo, "Using General-Purpose Serial-Link Manipulators for Laparoscopic Surgery with Moving Remote Center of Motion," *Int. J. Med. Robot. Res.*, vol. 01, no. 04, p. 1650007, 2016.

[103] T. Kröger and F. M. Wahl, "Online Trajectory Generation: Basic Concepts for Instantaneous Reactions to Unforeseen Events," *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 94–111, 2010.

[104] P. Beeson and B. Ames, "TRAC-IK: An open-source library for improved solving of generic inverse kinematics," in *Proc. IEEE-RAS Int. Conf. Human. Robot.*, 2015, pp. 928–935. [Online]. Available: ieeexplore.ieee.org

[105] R. P. Young and R. G. Marteniuk, "Acquisition of a multi-articular kicking task: Jerk analysis demonstrates movements do not become smoother with learning," *Hum. Mov. Sci.*, vol. 16, no. 5, pp. 677–701, 1997.

[106] J. Colan, J. Nakanishi, T. Aoyama, and Y. Hasegawa, "A Cooperative Human-Robot Interface for Constrained Manipulation in Robot-Assisted Endonasal Surgery," *Appl. Sci.*, vol. 10, no. 14, p. 4809, 2020.

[107] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg, and P. Abbeel, "Motion planning with sequential convex optimization and convex collision checking," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1251–1270, 2014.

[108] R. Bonalli, A. Cauligi, A. Bylard, and M. Pavone, "GuSTO: Guaranteed sequential trajectory optimization via sequential convex programming," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 6741–6747.

[109] R. Fletcher, *Practical Methods of Optimization*. John Wiley & Sons, 1987.

[110] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer Science & Business Media, 2006.

[111] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[112] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Trans. Robot. Automat.*, vol. 13, no. 3, pp. 398–410, 1997.

[113] H. Azimian, R. V. Patel, and M. D. Naish, "On Constrained Manipulation in Robotics-Assisted Minimally Invasive Surgery," in *Proc. IEEE/EMBS Int. Conf. Biomed. Robot. Biomechatron.*, 2010, pp. 650–655.

[114] J. Sandoval, G. Poisson, and P. Vieyres, "A New Kinematic Formulation of the RCM Constraint for Redundant Torque-Controlled Robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2017, pp. 4576–4581.

[115] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The Pinocchio C++ library," in *Proc. IEEE/SICE Int. Sym. Syst. Integ.*, 2019, pp. 614–619.

[116] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: a software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, no. 1, pp. 1–36, 2019.

[117] Gurobi Optimization, LLC. Gurobi optimizer reference manual. Accesed on: Dec. 20, 2020. [Online]. Available: https://www.gurobi.com/

[118] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006.

[119] E. Rohmer, S. P. N. Singh, and M. Freese, "V-REP: a Versatile and Scalable Robot Simulation Framework," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2013, pp. 1321–1326.

List of Publications

(1) Journal

[1] Jacinto Colan, Jun Nakanishi, Tadayoshi Aoyama, and Yasuhisa Hasegawa. "A Cooperative Human-Robot Interface for Constrained Manipulation in Robot-Assisted Endonasal Surgery." *Applied Sciences*, vol. 10, no. 14, p. 4809, 2020.

[2] Jacinto Colan, Jun Nakanishi, Tadayoshi Aoyama, and Yasuhisa Hasegawa. "Optimization-based constrained trajectory generation for robot-assisted stitching in endonasal surgery." *Robotics*, vol. 10, no. 1, p. 27, 2021.

(2) International Conference

[1] Daisuke Uozumi, Jacinto Colan, Jun Nakanishi, Tadayoshi Aoyama, and Yasuhisa Hasegawa. "Usability Study on Hands-on User Interface for Neurosurgical Articulated Forceps: Joystick and Serial-link based Design." Proc. IEEE International Symposium on Micro-NanoMechatronics and Human Science (MHS), 2019.

[2] Keisuke Ohara, Jacinto Colan, Daisuke Uozumi, Jun Nakanishi, and Yasuhisa Hasegawa. "Development of a Precision-grip based Interface for 4-DoF Articulated Forceps." Proc. IEEE International Symposium on Micro-NanoMechatronics and Human Science, pp. 1–4, 2018.

[3] Jacinto Colan, Jun Nakanishi, Keisuke Ohara, Tadayoshi Aoyama, and Yasuhisa Hasegawa. "A concept of a user interface capable of intuitive operation of 4-DoF articulated forceps." Proc. IEEE International Symposium on Micro-NanoMechatronics and Human Science, pp. 1–3, 2017.

(3) Domestic Conference

[1] Daisuke Uozumi, Jacinto Colan, Yasuhisa Hasegawa, Tadayoshi Aoyama, and Jun Nakanishi. "Improving Dexterity for a Hands-on User Interface to Operate Articulated Forceps via Index Finger Movement with Seizing Grip." The Robotics and Mechatronics Conference (ROBOMECH), 2020. (In Japanese)

[2] Daisuke Uozumi, Keisuke Ohara, Jacinto Colan, Tadayoshi Aoyama, Jun Nakanishi, and Yasuhisa Hasegawa. "Development of Hands-on User Interface for Articulated Forceps Enabling Simultaneous Positioning and Tip Movement Operation." The Robotics and Mechatronics Conference (ROBOMECH), 2019. (In Japanese)

[3] Keisuke Ohara, Jacinto Colan, Daisuke Uozumi, Tadayoshi Aoyama, Jun Nakanishi, and Yasuhisa Hasegawa. "Development of pinch-based interface for multi-degree-of-freedom active forceps." The 36th Annual Conference of the Robotics Society of Japan, 2018. (In Japanese)

[4] Jacinto Colan, Yuichiro Sato, Jun Nakanishi, and Yasuhisa Hasegawa. "A Preliminary Study on Cooperative Force Control based Guidance for Accurate Pre–insertion Positioning of Surgical Instruments." The Robotics and Mechatronics Conference (ROBOMECH), 2017.

[5] Yuichiro Sato, Jacinto Colan, Jun Nakanishi, and Yasuhisa Hasegawa. "Design and Prototype of an Intuitive User Interface for Forceps with Multi-Degrees of Freedom." SICE SI Division Annual Conference, 2016. (In Japanese)

(4) Other publications

[1] Yasuhisa Hasegawa, Jacinto Colan, Daisuke Uozumi, Tadayoshi Aoyama, and Jun Nakanishi. "Smart arm: User interface." Special Issue on Bionic Humanoid, *System/Control/Information*, vol. 63, no. 10, pp. 424-430, 2019. (In Japanese)